Technische Universität Berlin

Forschungsberichte der Fakultät IV – Elektrotechnik und Informatik

Low- and High-Level Petri Nets with Individual Tokens

Tony Modica, Karsten Gabriel, Hartmut Ehrig,
Kathrin Hoffmann, Sarkaft Shareef, Claudia Ermel,
Ulrike Golas, Frank Hermann, Enrico Biermann

Bericht-Nr. 2009-13
ISSN 1436-9915

Low- and High-Level Petri Nets
with Individual Tokens

Tony Modica, Karsten Gabriel, Hartmut Ehrig,
Kathrin Hoffmann, Sarkaft Shareef, Claudia Ermel,
Ulrike Golas, Frank Hermann, Enrico Biermann

Bericht-Nr. 2009/13
ISSN 1436-9915

Technical Report 13/2009

Low- and High-Level Petri Nets with Individual Tokens*

Tony Modica, Karsten Gabriel, Hartmut Ehrig,
Kathrin Hoffmann, Sarkaft Shareef, Claudia Ermel,
Ulrike Golas, Frank Hermann, Enrico Biermann

E-Mail: {modica,kgabriel,ehrig,hoffmann,sarkaft,
lieske,ugolas,frank,enricob}@cs.tu-berlin.de

July 8, 2010

TFS - Fakultät IV - Elektrotechnik und Informatik
Technische Universität Berlin

*This work has been supported by the Integrated Graduate Program on Human-Centric Communication at TU Berlin and by the research project forMA$_l$NET (see http://tfs.cs.tu-berlin.de/formalnet/) of the German Research Council.

Abstract

In this article, we present a new variant of Petri nets with markings called "Petri nets with individual tokens", together with rule-based transformation following the double pushout approach. The most important change to former Petri net transformation approaches is that the marking of a net is no longer a "collective" set of tokens, but each each has an own identity leading to the concept of Petri nets with individual tokens. This allows us to formulate rules that can change the marking of a net arbitrarily without necessarily manipulating the structure. As a first main result that depends on nets with individual markings we show the equivalence of transition firing steps and the application of firing-simulating rules.

We define categories of low-level and of algebraic high-level nets with individual tokens, called PTI nets and AHLI nets, respectively, and relate them with each other and their collective counterparts by functors.

To be able to use the properties and analysis results of $\mathcal{M}$-adhesive HLR systems (formerly know as weak adhesive high-level replacement systems) we show in further main results that both categories of PTI nets and AHLI nets are $\mathcal{M}$-adhesive categories. By showing how to construct initial pushouts we also give necessary and sufficient conditions for the applicability of transformation rules in these categories, known as gluing condition in the literature.

Contents

1. Introduction

Petri nets are one of the main formalisms to describe and analyze concurrent processes [Rei85a]. To be able to utilize them in modeling of additional complex aspects, a lot of work has been done so far to extend the classical Petri nets or to integrate them with other formal techniques.

One of those challenging extensions is the manipulation of running processes, which has been formulated under the notion of adaptive workflows [vdABV+99]. In the past, several approaches have been proposed that tried to deal with this aspect of process modeling by using Petri nets that can change their firing behavior, e.g. self-modifying nets [Val78], mobile nets [AB09], recursive nets [HP00, HB08], and nets with dynamic transition refinement [KR07].

In this article, we concentrate on the approach to combine Petri nets with transformation rules based on graph transformation [EEPT06], which has been applied to adaptive workflows in [HME05, BDHM06]. This exploits the graph-like structure of Petri nets and allows us to formulate rules to change the structure of a Petri net [EHP+08]. In general, rule-based Petri net transformation can also be applied to Petri systems, i.e. Petri nets with a marking, which is especially useful for manipulation of processes at runtime.

Petri nets have been shown in [EEPT06, EHP+08] to be define a weak adhesive HLR category for the class $\mathcal{M}$ of all injective net morphisms. This allows us to apply all the results for adhesive HLR systems shown in [EEPT06] also for Petri net transformation systems. In this paper, we use the short notion "$\mathcal{M}$-adhesive category" for "weak adhesive HLR category". the concept of Petri systems leads to a category **PTSys** with morphisms allowing to increase the number of tokens on corresponding places. Unfortunately, $(\mathbf{PTSys}, \mathcal{M}_{inj})$ with the class $\mathcal{M}_{inj}$ of all injective morphisms is not $\mathcal{M}$-adhesive in contrast to $(\mathbf{PTSys}, \mathcal{M}_{strict})$, where $\mathcal{M}_{strict}$ is the class of injective morphisms where the number of tokens on corresponding places is equal. This, however, is an unpleasant restriction for the usability of the transformation approach, especially the firing of a transition cannot be simulated in a natural way by the application of a corresponding "transition rule".

To overcome this restriction we present a new Petri net formalism, called "Petri nets with individual tokens", together with a rule-based transformation approach that is almost equivalent to [EHP+08][1]. The most important change is that a net's marking is no longer a ("collective") sum of a monoid but a set of individuals. This allows us to formulate rules that can change the marking of a net arbitrarily so that, as major advantage, firing of nets can be modeled by rule applications.

As a main result we will show that Petri nets with individual tokens are $\mathcal{M}$-adhesive systems with all their nice properties.

With individual tokens, the category of Petri systems is even closer to typed attributed graphs, which opens an elegant way to simulate Petri net transformations with graph

[1] The approach in [EHP+08] follows the concept "Petri nets are monoids" [MM90] so that a net and its marking are represented as sets and monoids over these sets, rather than e.g. with an explicit flow relation. This makes it easier to handle categories of Petri nets and systems.

transformation tools [Syl09].

In Sect. 2, we introduce *low-level* Petri nets with individual tokens and define rule-based transformation. As a main result we show the equivalence of transition firing steps and the application of firing-simulating rules.

In Sect. 3, we lift the definitions and results of Sect. 2 to *high-level* nets that have data elements of an algebra [EM85] as tokens rather than low-level "black tokens". The first approach of inscripting Petri nets with algebraic terms was developed in [Rei91], however, we use the approach of [EPR93], which allows us to use an arbitrary algebra, as long as it complies to the given signature.[2]

Section 4 relates the Petri net classes of Sect. 2 and 3 with each other and with their collective counterparts by functors and we show a compatibility result for these functors.

We define $\mathcal{M}$-adhesive systems for both low and high-level Petri nets with individual tokens in Sect. 5 and discuss the results and some final remarks in the last sections.

1.1. The Notion of Individual Tokens

The term of *individual tokens* has been used in two senses for Petri nets so far:

1. It was mentioned first in [Rei85b], where it was used to describe "tokens that can be identified as individual objects". The main contribution of this article was to define markings as multisets of distinguished elements rather than amounts of indistinguishable black tokens. In the end, *individual tokens* in this context is a synonym for what by now is called *high-level tokens*. This is not our intended meaning of individuality, moreover we will consider both low-level and high-level Petri nets with individual tokens.

2. The other notion of token individuality has been coined in [vGP95] [3] as "individual token interpretation" of firing steps, which entitles the definition of processes in [GR83].

In [vG05], the author investigates the collective-individual dichotomy of firing steps, where under the individual approach firing sequences consider not only the number and value of tokens (as in the collective approach) but also the history, i.e. the origin transition, of tokens. [BMMS99] introduces a category of Petri nets to be interpreted with a functorial individual semantics.

Individual tokens in this article The Petri net approach we present in this article is related to 2., but in contrast, we will already introduce individual tokens in the (syntactical) definition of the Petri nets (and their category) themselves, so that we can exploit the individuality of tokens in the transformation approach, independently of the firing rule. The individual firing rule of [vG05] is very close to the one presented in the next sections, but it is still an interpretation of firing steps of a classic representation of Petri nets.

[2] Both articles consider algebraic specifications with equations instead of signatures (which can be understood as specifications without equations). In this article we take into account signatures only.

[3] Meanwhile, there is an updated version of this article: [vGP09].

The main purpose of our notion of nets with individual tokens is the possibility of formulating marking-changing rules for this kind of nets. We have demonstrated the practicability of such a transformation approach with modeling case studies on applications to Communication Spaces [BEE+09, Mod10, MEE+10].

2. Place/Transition Nets with Individual Tokens

In this section we treat low-level Petri nets and equip them with markings of individual tokens. After describing their firing behavior and introducing rules for the transformation of marked nets we show how special transformations correspond to transition firing steps.

2.1. Definition and Firing Behavior

With the definition of nets with individual tokens, we follow the concept "Petri nets are monoids" from [MM90]:

Definition 2.1 (Place/Transition Nets with Individual Tokens (PTI)).
We define a marked P/T net with individual tokens, short PTI net, as

$$NI = (P, T, pre, post, I, m),$$

where

- $N = (P, T,\ pre, post : T \to P^{\oplus})$ is a classical P/T net, where $P^{\oplus}$ is the free commutative monoid over P,
- I is the (possibly infinite) set of individual tokens of NI, and
- $m : I \to P$ is the marking function, assigning the individual tokens to the places.

Further, we introduce some additional notations:

- the environment of a transition $t \in T$ as $ENV(t) = PRE(t) \cup POST(t) \subseteq P$ with

$$PRE(t) = \{p \in P | pre(t)(p) \neq 0\},$$
$$POST(t) = \{p \in P | post(t)(p) \neq 0\},$$

Example. Figure 1 on the facing page shows the graphical representation of a PTI net with $I = \{x_1, y_1, x_2, x_3\}$, $m(x_1) = m(y_1) = p_1$, $m(x_2) = p_2$, $m(x_3) = p_3$.

Now that we have marked Petri nets, also called Petri systems, we have to define their behavior as firing steps. Because we have individual tokens, we have to consider a possible firing step in the context of a selection of tokens.

Definition 2.2 (Firing of PTI Nets).
A transition $t \in T$ in a PTI net

$$NI = (P, T, pre, post, I, m)$$

is *enabled* under a *token selection* (M, m, N, n), where

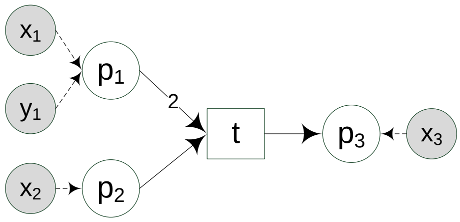

Figure 1: Example PTI net

- $M \subseteq I$,
- m is the token mapping of NI,
- N is a set with $(I \setminus M) \cap N = \emptyset$,
- $n : N \to P$ is a function,

if it meets the following *token selection condition*:

$$\left[\sum_{i \in M} m(i) = pre(t)\right] \wedge \left[\sum_{i \in N} n(i) = post(t)\right]$$

If an enabled t fires, the follower marking (I', m') is given by

$$I' = (I \setminus M) \cup N, \quad m' : I' \to P \text{ with } m'(x) = \begin{cases} m(x), & x \in I \setminus M \\ n(x), & x \in N \end{cases}$$

leading to $NI' = (P, T, pre, post, I', m')$ as the new PTI net in the *firing step* $NI \overset{t}{\rightarrowtail} NI'$ via (M, m, N, n).

Remark (Token Selection). The purpose of the token selection is to specify exactly which tokens should be consumed and produced in the firing step. Thus, $M \subseteq I$ selects the individual tokens to be consumed, and N contains the set of individual tokens to be produced. Clearly, $(I \setminus M) \cap N = \emptyset$ must hold because it is impossible to add an individual token to a net that already contains this token. m and n relate the tokens to their carrying places. It would be sufficient to consider only the restriction $m_{|M}$ here or to infer it from the net but as a compromise on symmetry and readability we denote m in the token selection.

For the next subsection we need a category of Petri systems.

Definition 2.3 (PTI Net Morphisms and Category **PTINets**).
Given two PTI nets $NI_i = (P_i, T_i, pre_i, post_i, I_i, m_i)$, $i \in \{1, 2\}$, a PTI net morphism is a triple of functions $f = (f_P : P_1 \to P_2, f_T : T_1 \to T_2, f_I : I_1 \to I_2) : NI_1 \to NI_2$, such that the following diagrams commute (componentwise for pre_i and $post_i$):

$$
\begin{tikzcd}
T_1 \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "post_1"'] \arrow[d, "f_T"'] \arrow[dr, phantom, "="] & P_1^{\oplus} \arrow[d, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2"] \arrow[r, shift right, "post_2"'] & P_2^{\oplus}
\end{tikzcd}
\qquad
\begin{tikzcd}
I_1 \arrow[r, "m_1"] \arrow[d, "f_I"'] \arrow[dr, phantom, "="] & P_1 \arrow[d, "f_P"] \\
I_2 \arrow[r, "m_2"] & P_2
\end{tikzcd}
$$

or, explicitly, that $f_P^{\oplus} \circ pre_1 = pre_2 \circ f_T$, $f_P^{\oplus} \circ post_1 = post_2 \circ f_T$, $f_P \circ m_1 = m_2 \circ f_I$. The category **PTINets** consists of all PTI nets as objects with all PTI net morphisms.

Remark. For a general PTI net morphism f we do not require that its token component f_I is injective in order to have pushouts, pullbacks, and a $\mathcal{M}$-adhesive category. But later we may require injectivity of f_I for rule and match morphisms and to have morphisms preserving firing behavior.

Fact 2.4 (Construction of Pushouts in **PTINets**).
Pushouts in **PTINets** are constructed componentwise in **PTNets** and **Sets**. So, (1) is a PO in **PTINets** iff (2) is a PO in **PTNets** and (3) is a PO in **Sets** with the components of the **PTINets** morphisms, where $m_3 : I_3 \to P_3$ is induced by PO object I_3 in the commuting cube below (whose front is the PO of place sets).

$$
\begin{tikzcd}
NI_0 \arrow[r, "f_1"] \arrow[d, "f_2"'] \arrow[dr, phantom, "(1)"] & NI_1 \arrow[d, "g_1"] \\
NI_2 \arrow[r, "g_2"'] & NI_3
\end{tikzcd}
$$

$$
\begin{tikzcd}
N_0 \arrow[r, "f_1'"] \arrow[d, "f_2'"'] \arrow[dr, phantom, "(2)"] & N_1 \arrow[d, "g_1'"] \\
N_2 \arrow[r, "g_2'"'] & N_3
\end{tikzcd}
\qquad
\begin{tikzcd}
I_0 \arrow[r, "f_{1I}"] \arrow[d, "f_{2I}"'] \arrow[dr, phantom, "(3)"] & I_1 \arrow[d, "g_{1I}"] \\
I_2 \arrow[r, "g_{2I}"'] & I_3
\end{tikzcd}
$$

$$
\begin{tikzcd}
I_0 \arrow[rr, "f_{1I}"] \arrow[dr, "m_0"] \arrow[dd, "f_{2I}"'] & & I_1 \arrow[dr, "m_1"] \arrow[dd, "g_{1I}"] & \\
& P_0 \arrow[rr, "f_{1P}"] \arrow[dd, "f_{2P}"] & & P_1 \arrow[dd, "g_{1P}"] \\
I_2 \arrow[rr, "g_{2I}"] \arrow[dr, "m_2"'] & & I_3 \arrow[dr, dashed, "m_3"] & \\
& P_2 \arrow[rr, "g_{2P}"'] & & P_3
\end{tikzcd}
$$

If f_{1X}, for $X \in \{P, T, I\}$, is injective then g_{2X} is as well; analogously for components of f_2 and g_1.

Example. Figure 2 on page 14 shows two pushouts in **PTINets**.

Fact 2.5 (Construction of Pullbacks in **PTINets**).
Pullbacks in **PTINets** along injective **PTINets** morphisms[4] are constructed componentwise in **PTNets** and **Sets**.

Consider a commutative square (1) in **PTINets** with g_1 having an injective net component g_1'. (1) is a PB in **PTINets** iff (2) is a PB in **PTNets** and (3) is a PB in **Sets** with the components of the **PTINets** morphisms, where $m_0 : I_0 \to P_0$ is induced by PB object P_0 in the commuting cube below (whose front is the PB of place sets).

[4] We require morphisms that are injective in order to obtain componentwise pullbacks in **PTINets**. See [EEPT06] for details.

$$
\begin{tikzcd}
NI_0 \arrow[r, "f_1"] \arrow[d, "f_2"'] \arrow[dr, phantom, "(1)"] & NI_1 \arrow[d, "g_1"] \\
NI_2 \arrow[r, "g_2"'] & NI_3
\end{tikzcd}
\qquad
\begin{tikzcd}
I_0 \arrow[rr, "f_{1I}"] \arrow[dr, dashed, "m_0"] \arrow[dd, "f_{2I}"'] & & I_1 \arrow[dr, "m_1"] \arrow[dd, "g_{1I}"] & \\
& P_0 \arrow[rr, "f_{1P}"] \arrow[dd, "f_{2P}"] & & P_1 \arrow[dd, "g_{1P}"] \\
I_2 \arrow[rr, "g_{2I}"] \arrow[dr, "m_2"'] & & I_3 \arrow[dr, "m_3"] & \\
& P_2 \arrow[rr, "g_{2P}"'] & & P_3
\end{tikzcd}
$$

$$
\begin{tikzcd}
N_0 \arrow[r, "f_1'"] \arrow[d, "f_2'"'] \arrow[dr, phantom, "(2)"] & N_1 \arrow[d, "g_1'"] \\
N_2 \arrow[r, "g_2'"'] & N_3
\end{tikzcd}
\qquad
\begin{tikzcd}
I_0 \arrow[r, "f_{1I}"] \arrow[d, "f_{2I}"'] \arrow[dr, phantom, "(3)"] & I_1 \arrow[d, "g_{1I}"] \\
I_2 \arrow[r, "g_{2I}"'] & I_3
\end{tikzcd}
$$

Example. The pushouts in Fig. 2 on page 14 are pullbacks, too.

2.2. Transformation of PTI Nets

This section is about rule-based transformation of PTI nets. We use the double pushout approach, which has also been used in [EHP+08] and stems from [EEPT06]. We are going to characterize the applicability of rules at some match by initial pushouts.

Definition 2.6 (PTI Transformation Rules).
A PTI transformation rule is a span of injective **PTINets** morphisms

$$\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R).$$

Remark. In contrast to [EHP+08], rule morphisms for PTI rules are not required to be marking-strict in order to obtain an $\mathcal{M}$-adhesive category (see Sect. 5 on page 41). This allows to arbitrarily change the marking of a PTI net by applying rules with corresponding places and individual tokens in L and R.

Definition 2.7 (PTI Transformation).
Given a PTI transformation rule $\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ and a PTI net NI_1 with a PTI net morphism $f : L \to NI_1$, called the match, a direct PTI net transformation $NI_1 \overset{\varrho,f}{\Longrightarrow} NI_2$ from NI_1 to the PTI net NI_2 is given by the following double-pushout diagram (DPO) diagram in the category **PTINets**:

$$
\begin{tikzcd}
L \arrow[d, "f"'] \arrow[dr, phantom, "(PO_1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(PO_2)"] & R \arrow[d, "f^*"] \\
NI_1 & NI_0 \arrow[l] \arrow[r] & NI_2
\end{tikzcd}
$$

To be able to decide whether a rule is applicable at a certain match, we formulate a gluing condition for PTI nets, such that there exists a pushout complement of the left rule morphisms and the match if (and only if) they fulfill the gluing condition. The correctness of the gluing condition is shown via a proof on initial pushouts over matches, according to [EEPT06].

Definition 2.8 (Gluing Condition in **PTINets**).
Given PTI nets K, L, and NI and PTI morphisms $l : K \to L$ and $f : L \to NI$. We define the set of identification points[5]

$$IP = IP_P \cup IP_T \cup IP_I$$

with

- $IP_P = \{x \in P_L \mid \exists x' \neq x : f_P(x) = f_P(x')\}$,
- $IP_T = \{x \in T_L \mid \exists x' \neq x : f_T(x) = f_T(x')\}$,
- $IP_I = \{x \in I_L \mid \exists x' \neq x : f_I(x) = f_I(x')\}$,

the set of dangling points[6]

$$DP = DP_T \cup DP_I$$

with

- $DP_T = \{p \in P_L \mid \exists t \in T_{NI} \setminus f_T(T_L) : f_P(p) \in ENV(t)\}$,
- $DP_I = \{p \in P_L \mid \exists i \in I_{NI} \setminus f_I(I_L) : f_P(p) = m_{NI}(i)\}$,

and the set of gluing points[7]

$$GP = l_P(P_K) \cup l_T(T_K) \cup l_I(I_K)$$

We say that l and f satisfy the gluing condition if $IP \cup DP \subseteq GP$. Given a PTI rule $\varrho = (L \xleftarrow{l} K \xrightarrow{r} R)$, we say ϱ and f satisfy the gluing condition iff l and f satisfy the gluing condition.

$$\begin{tikzcd} L \arrow[d, "f"'] & K \arrow[l, "l"'] \arrow[r, "r"] & R \\ NI & & \end{tikzcd}$$

In order to construct the initial pushout for a match $f : L \to NI$, we define the *boundary* over the match f, which is the minimal subnet containing all places, transitions, and individual tokens that must not be deleted by the application of a rule with left hand side L such that there exists a pushout complement.

Definition 2.9 (Boundary in **PTINets**).
Given a morphism $f : L \to NI$ in **PTINets**. The boundary of f is a PTI net

$$B = (P_B, T_B, pre_B, post_B, I_B, m_B)$$

with

[5] That is, all elements in L that are mapped non-injectively by f.
[6] That is, all places in L that would leave a dangling arc, if deleted.
[7] That is, all elements in L that have a preimage in K.

- $P_B = DP_T \cup DP_I \cup IP_P \cup P_{IP_T} \cup P_{IP_I}$
- $P_{IP_T} = \{p \in P_L \mid \exists t \in IP_T : p \in ENV(t)\}$
- $P_{IP_I} = \{p \in P_L \mid \exists i \in IP_I : p = m_L(i)\}$
- $T_B = IP_T$
- $pre_B(t) = pre_L(t)$
- $post_B(t) = post_L(t)$
- $I_B = IP_I$
- $m_B(i) = m_L(i)$

together with an inclusion $b : B \to L$.

Well-definedness.

$pre_B, post_B : T_B \to P_B^{\oplus}$:
The well-definedness follows from the well-definedness of Definition A.9 and the fact that B has the same set of transitions as the boundary in A.9 and the set of places in Definition A.9 is a subset of P_B.

$m_B : I_B \to P_B$:
Let $i \in I_B$. Then we have $i \in IP_I$ and hence

$$m_B(i) = m_L(i) \in P_{IP_I} \subseteq P_B$$

$b : B \to L$:
We obtain an inclusion morphism $b : B \to L$ from the fact that $pre_B, post_B$ and m_B are restrictions of the respective functions in L.

□

The following facts about the gluing condition and initial pushouts hold in all $\mathcal{M}$-adhesive categories $(\mathbf{PTINets}, \mathcal{M})$ whose morphism class $\mathcal{M}$ of monomorphisms contains at least inclusions (for concrete instantiations see Sect. 5 on page 41). The next fact completes the construction of initial pushouts for matches and shows that the construction of the boundary in Def. 2.9 on the preceding page complies to the categorical notion of boundaries in Def. A.1 on page 49.

Fact 2.10 (Initial Pushout in **PTINets**).
Given a morphism $f : L \to NI$ in **PTINets**, the boundary B of f and the PTI net

$$C = (P_C, T_C, pre_C, post_C, I_C, m_C)$$

with

- $P_C = (P_{NI} \setminus f_P(P_L)) \cup f_P(b_P(P_B))$
- $T_C = (T_{NI} \setminus f_T(T_L)) \cup f_T(b_T(T_B))$
- $I_C = (I_{NI} \setminus f_I(I_L)) \cup f_I(b_I(I_B))$
- $pre_C(t) = pre_{NI}(t)$
- $post_C(t) = post_{NI}(t)$
- $m_C(i) = m_{NI}(i)$

Then diagram (1) where $g := f|_B$ is an initial pushout in **PTINets**.

$$\begin{tikzcd} B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] \\ C \arrow[r, "c"'] & NI \end{tikzcd}$$

Proof. See section B.1 on page 63. □

The following two facts show the correspondence between the gluing condition in **PTINets** and the categorical gluing condition (see Def. A.2 on page 49), which is a necessary and sufficient condition for the (unique) existence of pushout complements in all $\mathcal{M}$-adhesive categories.

Fact 2.11 (Characterization of Gluing Condition in **PTINets**)**.**
Let $l : K \to L$ and $f : L \to NI$ be morphisms in **PTINets** with $l \in \mathcal{M}$.
The morphisms l and f satisfy the gluing condition in **PTINets** if and only if they satisfy the categorical gluing condition.

Proof. See section B.2 on page 67. □

Fact 2.12 (Gluing Condition for PTI Transformation)**.**
Given a PTI rule $\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ and a match $f : L \to NI$ into a PTI net $NI = (N, I, m : I \to P_{NI})$. The rule ϱ is applicable on match f, i.e. there exists a (unique) pushout complement NI_0 in the diagram below, iff ϱ and f satisfy the gluing condition in **PTINets**.

$$\begin{tikzcd} L \arrow[d, "f"'] \arrow[dr, phantom, "(PO)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, dashed, "f'"] & R \\ NI & NI_0 \arrow[l, dashed] & \end{tikzcd}$$

Proof. By Fact 2.11 ϱ and f satisfy the gluing condition in **PTINets** if and only if ϱ and f satisfy the categorical gluing condition, which by Fact A.3 is a necessary and sufficient condition for the (unique) existence of the pushout complement NI_0. □

2.3. Correspondence of Transition Firing and Rules

Now that we can manipulate a net's marking with rules, we have a look at rules that simulate firing steps for a certain transition under a token selection. We show that firing of a transition corresponds to a canonical application of a special rule construction, called transition rules. With this correspondence we can easily show that token-injective PTI morphisms preserve firing steps.

Definition 2.13 (PTI Transition Rules)**.**
We define the *transition rule* for a transition $t \in T$ enabled under a token selection $S = (M, m, N, n)$ in a PTI net $NI = (P, T, pre, post, I, m)$ as the rule

$$\varrho(t, S) = (L_t \overset{l}{\leftarrow} K_t \overset{r}{\rightarrow} R_t), \text{ with}$$

- the common fixed net structure $PN_t = (P_t, \{t\}, pre_t, post_t)$, where $P_t = ENV(t)$, $pre_t(t) = pre(t)$ and $post_t(t) = post(t)$,
- $L_t = (PN_t, M, m_t : M \to P_t)$, with $m_t(x) = m(x)$,
- $K_t = (PN_t, \emptyset, \ \emptyset : \emptyset \to P_t)$,
- $R_t = (PN_t, N, n_t : N \to P_t)$, with $n_t(x) = n(x)$,
- l, r being the obvious inclusions on the rule nets.

m_t and n_t are well-defined because t is enabled under S: The token selection condition implies that $\forall x \in M : m(x) \in PRE(t)$ and due to the construction of PN_t we have $PRE(t) \subseteq ENV(t) = P_t$. The argument for n_t works analogously.

Note that t is enabled under S in L_t.

Example. Figure 2 shows a PTI transition rule $\varrho(t, S) = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ for transition t in NI.

Definition 2.14 (Canonical DPO Transformation of PTI Nets)**.**
We call a direct transformation $NI_1 \overset{\varrho,f}{\Longrightarrow} NI_2$ by rule $\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ with l, r being inclusions *canonical* if

- f_I is injective,
- the morphisms in the span $(NI_1 \leftarrow NI_0 \rightarrow NI_2)$ of the DPO transformation diagram below are inclusions, and
- $I_2 = I_0 \cup (I_R \setminus r(I_K))$.

$$\begin{tikzcd}
L \arrow[d, "f"'] \arrow[dr, phantom, "(PO_1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(PO_2)"] & R \arrow[d, "f^*"] \\
NI_1 & NI_0 \arrow[l] \arrow[r] & NI_2
\end{tikzcd}$$

Remark. For each rule ϱ being applied to some PTI net NI at a token-injective match f, there exists a canonical transformation diagram, which is just the particular equivalent DPO diagram with "as less isomorphic changes as possible".

Because of the injectivity of the rule morphisms the lower span in the diagram is already injective as well. To construct the canonical transformation diagram we first regard the last condition on I_2, which demands that $(I_R \setminus r(I_K)) \cap I_0 = \emptyset$ because of pushout PO_2. For this, it is sufficient to replace the set of tokens in R that are created by the rule, i.e. $I_R \setminus r(I_K)$, such that is disjoint to the tokens preserved by the rule, i.e. $(I_1 \setminus f(I_L)) \cup f(l(I_K))$.[8] With this we now can simply replace arbitrary NI_0 and NI_2, being some pushout complement object of PO_1 and pushout object of PO_2, with nets such that the lower span morphisms become inclusions.

Example. The diagram in Fig. 2 shows the two pushouts in **PTINets** resulting from applying the PTI transition rule $\varrho(t,S) = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ to the net NI with identical token morphism component. Moreover, this transformation is canonical.

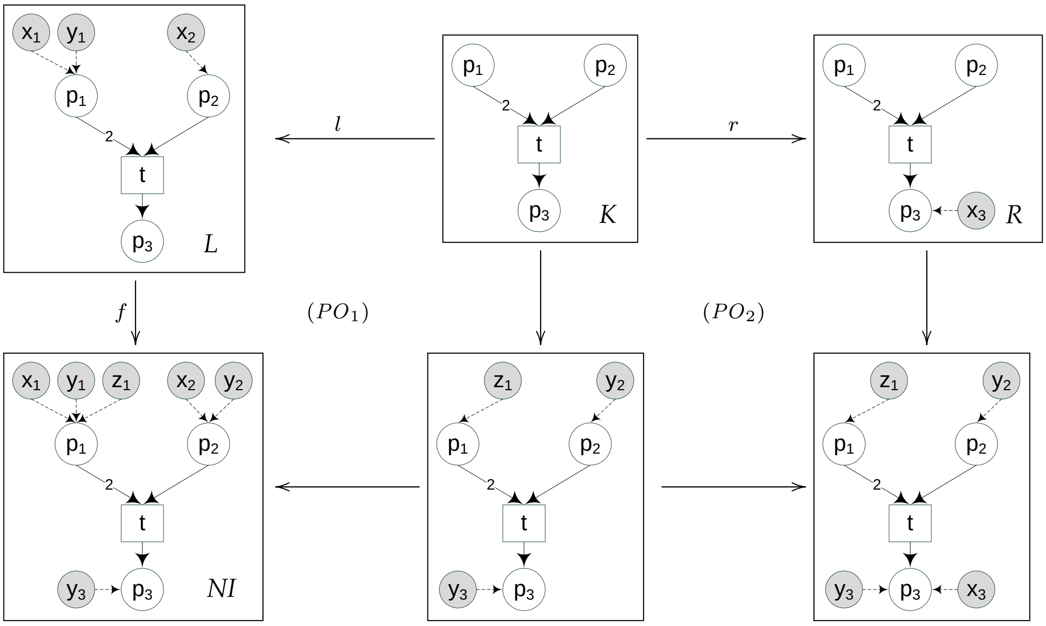

Figure 2: Canonical direct DPO-transformation in **PTINets** simulating a firing step

Theorem 2.1 (Equivalence of Canonical Transformations and Firing of PTI Nets)**.**

1. Each firing step $NI \overset{t}{\rightarrowtail} NI'$ *via a token selection* $S = (M, m, N, n)$ *corresponds to a canonical direct transformation* $NI \overset{\varrho(t,S),f}{\Longrightarrow} NI'$ *via the transition rule* $\varrho(t,S)$*, matched by the inclusion match* $f : L_{\varrho(t,S)} \to NI$.

[8] This modification of the rule is passable since changing the tokens in R does not affect the firing behavior of NI_2 (up to the token selections) nor the applicability of the rule.

2. Each canonical direct transformation $NI \xRightarrow{\varrho(t,S),f} NI_1$*, via some transition rule* $\varrho(t,S) = (L \xleftarrow{l} K \xrightarrow{r} R)$ *with* $t \in T_{NI}$ *and token selection* $S = (M, m, N, n)$*, and a token-injective match* $f : L \to NI$*, implies that the transition* $f_T(t)$ *is enabled in NI under* $\left(f_I(M), \left(f_P \circ m \circ f_I^{-1}\right), N, \left(f_P^* \circ m_1 \circ f_I^{*-1}\right)\right)$ *with firing step* $NI \xrightarrow{f_T(t)} NI_1$*.*

Proof.
1. Given the firing step $NI \xrightarrow{t} NI'$ via $S = (M, m, N, n)$, the canonical direct transformation of the transition rule $\varrho(t,S) = (L \xleftarrow{l} K \xrightarrow{r} R)$ is $NI \xRightarrow{\varrho(t,S),f} NI_1$ as in the DPO diagram in Fig. 3. The match f, d, and d' are inclusions.

$$\begin{tikzcd}
L = (PN_t, M, m_t) \arrow[d, "f"'] \arrow[r, phantom, "(PO_1)" near start, shift right=4ex] & K = (PN_t, \emptyset, \emptyset) \arrow[l, hook', "l"'] \arrow[r, hook, "r"] \arrow[d] \arrow[r, phantom, "(PO_2)", shift right=4ex] & R = (PN_t, N, n_t) \arrow[d, "f^*"] \\
NI = (PN, I, m) & NI_0 = (PN, I_0, m_0) \arrow[l, hook', "d"] \arrow[r, hook, "d'"'] & NI_1 = (PN, I_1, m_1)
\end{tikzcd}$$

Figure 3: DPO diagram for canonical direct transformation of NI with $\varrho(t,S)$ in **PTINets**

According to Fact 2.4 on page 8, we have $I_0 = I \setminus M$ and $I_1 = I_0 \cup (N \setminus \emptyset)$ as in the DPO diagram of the **Sets** components in Fig. 4. The firing step condition $(I \setminus M) \cap N = \emptyset$ grants us the last condition for canonical transformations $I_0 \cup (N \setminus r(\emptyset)) = I_1$.

$$\begin{tikzcd}
M \arrow[d, "f_I"'] \arrow[r, phantom, "(PO_1)", shift right=4ex] & \emptyset \arrow[l, hook'] \arrow[r, hook] \arrow[d] \arrow[r, phantom, "(PO_2)", shift right=4ex] & N \arrow[d, "f_I^*"] \arrow[dr, "n_t"] & \\
I \arrow[dr, "m"'] & I_0 = (I \setminus M) \arrow[l, hook', "d_I"] \arrow[r, hook, "d_I'"] \arrow[d, dashed, "m_0 = m \circ d_I"] & I_1 = (I \setminus M) \cup N \arrow[dr, dashed, "m_1"'] & P_t \arrow[d, "f_P^* = f_P"] \\
& P_{PN} \arrow[rr, equal, "id"'] & & P_{PN}
\end{tikzcd}$$

Figure 4: DPO diagram in **Sets** for the token components of the transformation in Fig. 3

For m_1 as induced morphism for the pushout object I_1 follows that

$$m_1(x) = \begin{cases} m_0(x) = m(x) \text{ for } x \in I \setminus M \\ n_t(x) = n(x) \text{ for } x \in N \end{cases}$$

hence $I_1 = I', m_1 = m'$ and $NI_1 = NI'$, according to Def. 2.2 on page 6.

Remark. f_I being injective is not only sufficient for the existence of (PO_1), but also necessary, because $I_K = \emptyset$ (see Fact 2.12 on page 12).

2. Given a canonical direct transformation $NI \xRightarrow{\varrho(t,S),f} NI_1$ as in the DPO diagrams in Figs. 3 and 4, $f_T(t) \in T_{NI}$ is enabled under

$$\left(f_I(M), \left(f_P \circ m \circ f_I^{-1}\right), N, \left(f_P^* \circ m_1 \circ f_I^{*-1}\right)\right)$$

if

1. $f_I(M) \subseteq I$,
2. $\left(f_P^* \circ m_1 \circ f_I^{*-1}\right) : N \to P_{PN}$,
3. $(I \setminus f_I(M)) \cap N = \emptyset$,
4. $\sum\limits_{i \in f_I(M)} \left(f_P \circ m \circ f_I^{-1}(i)\right) = pre_{NI}(f_T(t))$
5. $\sum\limits_{i \in N} \left(f_P^* \circ m_1 \circ f_I^{*-1}(i)\right) = post_{NI}(f_T(t))$
6. leading to the follower marking $I' = (I \setminus M) \cup N$
   with $m'(x) = \begin{cases} m_0(x) = m(x) \text{ for } x \in I \setminus M \\ m_1(x) = n(x) \text{ for } x \in N \end{cases}$

By construction of the transition rule $\varrho(t, S)$ and its application to NI we have 1., 2., and for 6. that $I' = I_1, m' = m_1$. The canonical transformation property grants us 3. It remains to show 4. and 5., i.e. that $f_I(M)$ and N represent the correct numbers of tokens in the environment of $f_T(t)$ to enable it:

$$
\begin{aligned}
& \sum_{i \in f_I(M)} f_P \circ m \circ f_I^{-1}(i) \\
&= \sum_{i \in M} f_P \circ m(i) && (f_I \text{ inj.}) \\
&= f_P^{\oplus} \sum_{i \in M} m_t(i) && (\forall i \in M : m_t(i) = m(i), \text{due to def. } \varrho(t, S)) \\
&= f_P^{\oplus} \circ pre_{PN_t}(t) && (t \text{ enabled in } L) \\
&= pre_{NI} \circ f_T(t) && (f \text{ PTI morph.})
\end{aligned}
$$

and analogously,

$$
\begin{aligned}
& \sum_{i \in N} f_P^* \circ m_1 \circ f_I^{*-1}(i) \\
&= \sum_{i \in N} f_P^* \circ m_1(i) && (f_I^*(N) = N) \\
&= f_P^{*\oplus} \sum_{i \in N} n_t(i) && (\forall i \in N : n_t(i) = m_1(i), \text{due to constr. } m_1) \\
&= f_P^{*\oplus} \circ pre_{PN_t}(t) && (t \text{ enabled in } L) \\
&= post_{NI} \circ f_T(t) && (f \text{ PTI morph.})
\end{aligned}
$$

□

The second item of the previous theorem covers all possible canonical direct transformations by any transition rule $\varrho(t, S)$ in an arbitrary net NI, without aasuming that $t \in NI$ or $M \subseteq I_{NI}$. For the special case that a transition rule $\varrho(t, S)$ is applied on an inclusion match, the second item reduces to the following corollary, which is more similar to the theorem's first item.

Corollary 2.2 (Equivalence of Canonical Transformations and Firing of PTI Nets).
Given a canonical transformation $NI \xRightarrow{\varrho(t,S),f} NI_1$ *such that the match* $f : L \to NI$ *is an inclusion, then* t *is enabled in* NI *under* S *with firing step* $NI \overset{t}{\rightarrowtail} NI_1$.

This follows directly from 2. of Theorem 2.1.

Theorem 2.3 (Token-injective PTI Net Morphisms preserve Firing Steps).
For each PTI net morphism $f : NI_1 \to NI_2$ *with injective* f_I *component and each firing step* $NI_1 \overset{t}{\rightarrowtail} NI_1'$ *there exists a firing step* $NI_2 \overset{f_T(t)}{\rightarrowtail} NI_2'$ *and a PTI net morphism* $f' : NI_1' \to NI_2'$ *(depicted in diagram (1) below).*

$$
\begin{tikzcd}
NI_1 \arrow[r, tail, "t"] \arrow[d, "f"'] \arrow[dr, phantom, "(1)"] & NI_1' \arrow[d, "f'"] \\
NI_2 \arrow[r, tail, "f_T(t)"'] & NI_2'
\end{tikzcd}
$$

Proof. Given $f : NI_1 \to NI_2$ and $NI_1 \overset{t}{\rightarrowtail} NI_1'$ via some S as above, we have by the first part of Theorem 2.1 on page 14 the canonical direct transformation given by the pushouts (1) and (2) with $\varrho(t, S) = (L \overset{l}{\leftarrow} K \overset{r}{\to} R)$ in Fig. 5.

$$
\begin{tikzcd}
L \arrow[d, "inc_L"'] \arrow[dr, phantom, "(1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(2)"] & R \arrow[d, "inc_R"] \\
NI_1 \arrow[d, "f"'] \arrow[dr, phantom, "(3)"] & NI_{D1} \arrow[l, "l'"'] \arrow[r, "r'"] \arrow[d] \arrow[dr, phantom, "(4)"] & NI_1' \arrow[d, "f'"] \\
NI_2 & NI_{D2} \arrow[l] \arrow[r] & NI_2'
\end{tikzcd}
$$

Figure 5: DPO diagrams for canonical direct transformation of NI_1 and NI_2 with $\varrho(t, S)$

Note that $f : NI_1 \to NI_2$ satisfies the gluing condition w.r.t this rule and f, because $l'_P = id_{P_1}, l'_T = id_{T_1}$[9] and none of the individuals of NI_1 is an identification point ($IP_I = \emptyset$) due to injectivity of f_I. This allows to construct the canonical transformation with extended rule $NI_1 \overset{l'}{\leftarrow} NI_{D1} \overset{r'}{\to} NI_1'$ along pushouts (3) and (4). Hence, also $(1 + 3)$ and $(2 + 4)$ are pushouts of a canonical direct transformation via $\varrho(t, S)$. The second part of Theorem 2.1 on page 14 implies $NI_2 \overset{f_T(t)}{\rightarrowtail} NI_2'$. $\square$

[9] This means that all places and transitions of NI_1 are gluing points.

Remark (Firing-preserving diagrams for t correspond to an extension diagram for $\varrho(t,S)$).
The diagram in Fig. 5 on the previous page corresponds to an extension diagram for rule $\varrho(t,S)$ and f (as depicted below) because $(1)-(4)$ are pushouts.

$$\begin{tikzcd} NI_1 \arrow[r, Rightarrow, "{\varrho(t,S),inc}"] \arrow[d, "f"'] & NI'_1 \arrow[d, "f'"] \\ NI_2 \arrow[r, Rightarrow, "{\varrho(t,S),f\circ inc}"] & NI'_2 \end{tikzcd}$$

3. Algebraic High-Level Nets with Individual Tokens

In this section, we lift the results of the previous section to high-level nets whose tokens represent values of an algebra to a signature [EM85]. The rule-based transformation of *collective* algebraic high-level nets from [PER95] is the foundation for the approach presented in the following.

3.1. Definition and Firing Behavior

Definition 3.1 (Algebraic High-Level Nets with Individual Tokens).
We define a marked AHL net with individual tokens, short AHLI net, as

$$ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m),$$

where

- $AN = (\Sigma, P, T, pre, post, cond, type, A)$ is a classical AHL net with
  - signature $\Sigma = (S, OP, X)$ of sorts S, operation symbols OP and variables $X = (X_s)_{s\in S}$,
  - sets of places P and transitions T,
  - $pre, post : T \to (T_{OP}(X) \otimes P)^{\oplus}$[10], defining the transitions' pre- and postdomains,
  - $cond : T \to \mathcal{P}_{fin}(Eqns(S, OP, X))$ assigning a finite set of Σ-equations (L, R, X) as firing conditions to each transition,
  - $type : P \to S$ typing the places of the signature's sorts,
  - a Σ-algebra A,
- I is the (possibly infinite) set of individual tokens of ANI, and
- $m : I \to A \otimes P$ is the marking function, assigning the individual tokens to the data elements on the places. $m(I)$ defines the actual set of data elements on the places of ANI. m does not have to be injective.

Further, we introduce some additional notations:

[10] $T_{OP}(X) \otimes P = \{(t,p) \in T_{OP}(X) \times P | t \in T_{OP,type(p)}(X)\}$, i.e the pairs where term t is of sort $type(p)$.

- $Var(t) \subseteq X$ is the set of variables occurring in equations and on the environment arcs of t,
- $CP = (A \otimes P) = \{(a,p) \in A \times P | a \in A_{type(p)}\}$ as the set of consistent value/place pairs,
- $CT = \{(t, asg) \in T \times (Var(t) \to A) | \forall (L, R, X) \in cond(t) : \overline{asg}(L) = \overline{asg}(R)\}$ as the set of consistent transition assignments, i.e. all firing conditions of t are valid when evaluated with the variable assignment asg[11],
- $ENV(t) = PRE(t) \cup POST(t) \subseteq (T_{OP}(X) \otimes P)$ as the environment of a transition $t \in T$ where

$$PRE(t) = \{(term, p) \in (T_{OP}(X) \otimes P) | pre(t)(term, p) \neq 0\}$$
$$POST(t) = \{(term, p) \in (T_{OP}(X) \otimes P) | post(t)(term, p) \neq 0\}$$

- $ENV_P(t) = \pi_P(ENV(t)) \subseteq P$ the place environment of t,
- $pre_A, post_A : CT \to CP^{\oplus}$, defined by

$$pre_A(t, asg) = (\overline{asg} \otimes id_P)^{\oplus} (pre(t)),$$
$$post_A(t, asg) = (\overline{asg} \otimes id_P)^{\oplus} (post(t))$$

  Similarly, we define the sets[12]

$$PRE_A(t, asg) = \{(a,p) \in (A \otimes P) | pre_A(t, asg)(a,p) \neq 0\}$$
$$POST_A(t, asg) = \{(a,p) \in (A \otimes P) | post_A(t, asg)(a,p) \neq 0\}$$

  We can express e.g. the concrete required *number* of a token (a,p) for t to fire under assignment asg with $pre_A(t, asg)(a,p)$ by interpreting the monoid $pre_A(t, asg)$ as a function $CP \to \mathbb{N}$. Similarly, we get the produced *number* of (a,p) with $post_A(t, asg)(a,p)$.

Remark (Individual tokens vs. classical algebraic data tokens). Each AHLI net with individual token marking (I, m) can be interpreted as an AHL net with marking

$$M = \sum_{(a,p) \in A \otimes P} |m^{-1}(a,p)|(a,p) = \sum_{i \in I} m(i)$$

where $|m^{-1}(a,p)|$ denotes the cardinality of individual tokens in I that m maps to (a,p).

In AHL nets, the tokens are of the form (a,p), s.t. they have already a kind of identity, depending on their data values. The main difference to AHLI nets is that we can distinguish tokens of the same algebraic value on the same place. Moreover, the individuals equip data tokens with identities. When firing a transition, we now can relate the input and output tokens so that a token's history can be traced along the firing steps.

[11] where $\overline{asg} : T_{OP}(X) \to A$ is the evaluation of Σ-terms over variables in X to values in A. Technically, $\overline{asg} = xeval(asg)_A$ results from a free construction over asg.

[12] Obviously, these sets are the same as $(\overline{asg} \otimes id_P) \circ PRE(t)$ and $(\overline{asg} \otimes id_P) \circ POST(t)$, respectively.

Example. Figure 6 shows the graphical representation of an AHLI net with

- signature $\Sigma = (\{s_1, s_2, s_3\},\ \{t_{11} :\to s_1, t_{12} :\to s_1, t_2 :\to s_2, t_3 :\to s_3\})$,
- algebra carrier sets $A_{s_1} = \{a_1, b_1, c_1\}, A_{s_2} = \{a_2, b_2\}, A_{s_3} = \{a_3, b_3, c_1\}$
- $pre(t) = (t_{11}, p_1) \oplus (t_{12}, p_1) \oplus (t_2, p_2), post(t) = (t_3, p_3)$,
- $type(p_1) = s_1, type(p_2) = s_2, type(p_3) = s_3$,
- $cond(t) = \emptyset$,
- $I = \{x_1, y_1, x_2, x_3\},\ m(x_1) = m(y_1) = (a_1, p_1), m(x_2) = (a_2, p_2), m(x_3) = (a_3, p_3)$,

Note that the algebraic value of an individual token is given next to the dashed arc to its carrying place. In the following, if a transition has an empty set of conditions we just denote the transition name without an explicit $\emptyset$ below.

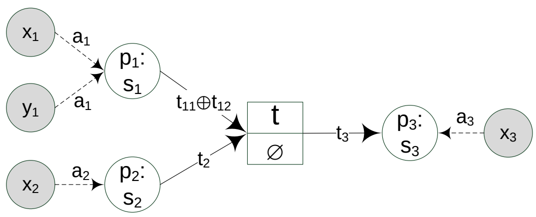

Figure 6: Example AHLI net

Similar as for low-level PTI nets, we now define firing steps for a transition and a token selection. In addition, we have to take into account assignments evaluating the variables on the arcs and in the transition conditions to algebra values.

Definition 3.2 (Firing of AHLI nets)**.**
A consistent transition assignment $(t, asg) \in CT$ for an AHLI net

$$ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m)$$

is *enabled* under a *token selection* (M, m, N, n), where

- $M \subseteq I$,
- m is the token mapping of ANI,
- N is a set with $(I \setminus M) \cap N = \emptyset$,
- $n : N \to A \otimes P$ is a function,

if it meets the following *token selection condition*:

$$\sum_{i \in M} m(i) = pre_A(t, asg) \wedge \sum_{i \in N} n(i) = post_A(t, asg)$$

If such an *asg*-enabled t fires, the follower marking (I', m') is given by

$$I' = (I \setminus M) \cup N, \quad m' : I' \to A \otimes P \text{ with } m'(x) = \begin{cases} m(x), & x \in I \setminus M \\ n(x), & x \in N \end{cases}$$

leading to $ANI' = (AN, I', m')$ as the new AHLI net in the *firing step* $ANI \xrightarrow{t,asg} ANI'$ via (M, m, N, n).

Remark (Token Selection). The purpose of the token selection is to specify exactly which tokens should be consumed and produced in the firing step. Thus, $M \subseteq I$ selects the individual tokens to be consumed, and N contains the set of individual tokens to be produced. Clearly, $(I \setminus M) \cap N = \emptyset$ must hold because it is impossible to add an individual token to a net that already contains this token. m and n relate the tokens to their place/value pairs. It would be sufficient to consider only the restriction $m_{|M}$ here or to infer it from the net but as a compromise on symmetry and readability we denote m in the token selection.

As a preparation for the transformation in the next subsection, we define a category of AHLI nets.

Definition 3.3 (AHLI Net Morphisms and Category **AHLINets**).
Given two AHLI nets $ANI_i = (\Sigma_i, P_i, T_i, pre_i, post_i, cond_i, type_i, A_i, I_i, m_i)$, $i \in \{1, 2\}$, an AHLI net morphism $f : ANI_1 \to ANI_2$ is a pentuple

$$f = (f_\Sigma : \Sigma_1 \to \Sigma_2, f_P : P_1 \to P_2, f_T : T_1 \to T_2, f_A : A_1 \to V_{f_\Sigma}(A_2), f_I : I_1 \to I_2)$$

such that the following diagrams commute (componentwise for pre_i and $post_i$)[13]:

[13] V_{f_Σ} is the forgetful functor induced by signature homomorphism f_Σ, such that $f_A : A_1 \to V_{f_\Sigma}(A_2)$ is a generalized Σ_1-homomorphism. $f_\Sigma^\#$ is the extension of f_Σ to terms and equations.

$$\begin{tikzcd}
\mathcal{P}_{fin}(Eqns(\Sigma_1)) \arrow[d, "\mathcal{P}_{fin}(f_\Sigma^\#)"'] & T_1 \arrow[l, "cond_1"'] \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "post_1"'] \arrow[d, "f_T"'] & (T_{OP_1}(X_1) \otimes P_1)^\oplus \arrow[d, "(f_\Sigma^\# \otimes f_P)^\oplus"] \\
\mathcal{P}_{fin}(Eqns(\Sigma_2)) & T_2 \arrow[l, "cond_2"] \arrow[r, shift left, "pre_2"] \arrow[r, shift right, "post_2"'] & (T_{OP_2}(X_2) \otimes P_2)^\oplus
\end{tikzcd}$$

$$\begin{tikzcd}
P_1 \arrow[r, "type_1"] \arrow[d, "f_P"'] & S_1 \arrow[d, "f_\Sigma"'] & I_1 \arrow[r, "m_1"] \arrow[d, "f_I"'] & A_1 \otimes P_1 \arrow[d, "f_A \otimes f_P"] \\
P_2 \arrow[r, "type_2"'] & S_2 & I_2 \arrow[r, "m_2"] & A_2 \otimes P_2
\end{tikzcd}$$

For a transition assignment $(t, asg : X_1 \to A_1)$ we call

$$asg_f = f_A \circ asg \circ f^{-1}_{\Sigma|Var(t)} : Var(f_T(t)) \to A_2$$

the *translation* of asg along f if f_Σ is bijective on the variables in $Var(t)$. Actually, for all transitions $t \in T_1$ all $f_{\Sigma|Var(t)} : Var(t) \to Var(f_T(t))$ are already surjective because of the commutativity of the diagrams above. So it is sufficient to demand injectivity for $f_{\Sigma|Var(t)}$ such that asg_f is well-defined.

The category **AHLINets** consists of all AHLI nets as objects and all AHLI net morphisms.

Remark (Generalized algebra morphisms). Because $V_{f_\Sigma}(A_2)$ just forgets some carrier sets of A_2 if we considered them as a family of sets, we may use f_A also with A_2 as codomain omitting the postponed family of identities $(\iota_s : (V_{f_\Sigma}(A_2))_s \to A_{2,f_\Sigma(s)})_{s\in S}$. Moreover, in the following constructions the algebra parts of the pushout cospan (and the pullback span resp.) result from general constructions in the Grothendieck construct with objects $(\Sigma, A \in \mathbf{Alg}(\mathbf{\Sigma}))$ and generalized algebra morphisms. See [EBO92, TBG91] for amalgamation of algebras and limits/colimits in Grothendieck constructs and also [EM85, EOP06] for details on the usage of free functors.

Fact 3.4 (Construction of Pushouts in **AHLINets**).
Pushouts in **AHLINets** are constructed componentwise in **AHLNets** and **Sets**. So, (1) is a PO in **AHLINets** iff (2) is a PO in **AHLNets** and (3) is a PO in **Sets** with the components of the **AHLINets** morphisms, where $m_3 : I_3 \to A_3 \otimes P_3$ is induced by PO object I_3 in the commutating cube below (whose front's place components let the front commutate because of the PO in the net structure).

$$
\begin{tikzcd}
ANI_0 \arrow[r, "f_1"] \arrow[d, "f_2"'] \arrow[dr, phantom, "(1)"] & ANI_1 \arrow[d, "g_1"] \\
ANI_2 \arrow[r, "g_2"'] & ANI_3
\end{tikzcd}
\qquad
\begin{tikzcd}
I_0 \arrow[rr, "f_{1I}"] \arrow[dr, "m_0"] \arrow[dd, "f_{2I}"'] & & I_1 \arrow[dr, "m_1"] \arrow[dd, "g_{1I}" near start] & \\
& A_0 \otimes P_0 \arrow[rr, "f_{1A} \otimes f_{1P}" near start] \arrow[dd, "f_{2A} \otimes f_{2P}" near start] & & A_1 \otimes P_1 \arrow[dd, "g_{1A} \otimes g_{1P}"] \\
I_2 \arrow[rr, "g_{2I}"' near end] \arrow[dr, "m_2"'] & & I_3 \arrow[dr, dashed, "m_3"] & \\
& A_2 \otimes P_2 \arrow[rr, "g_{2A} \otimes g_{2P}"'] & & A_3 \otimes P_3
\end{tikzcd}
$$

$$
\begin{tikzcd}
AN_0 \arrow[r, "f'_1"] \arrow[d, "f'_2"'] \arrow[dr, phantom, "(2)"] & AN_1 \arrow[d, "g'_1"] \\
AN_2 \arrow[r, "g'_2"'] & AN_3
\end{tikzcd}
\quad
\begin{tikzcd}
I_0 \arrow[r, "f_{1I}"] \arrow[d, "f_{2I}"'] \arrow[dr, phantom, "(3)"] & I_1 \arrow[d, "g_{1I}"] \\
I_2 \arrow[r, "g_{2I}"'] & I_3
\end{tikzcd}
$$

If f_{1X}, for $X \in \{P, T, I\}$, is injective then g_{2X} is as well and similar for components of f_2 and g_1 and the other diagrams.

For the construction of pushouts in **AHLNets** we refer to [PER95].

Example. Figure 7 on page 30 shows two pushouts in **AHLINets**.

Fact 3.5 (Construction of Pullbacks in **AHLINets**).
Pullbacks in **AHLINets** along injective **AHLNets** morphisms[14] are constructed componentwise in **AHLNets** and **Sets**.

Consider a commutative square (1) in **AHLINets** with g'_1 being injective. (1) is a PB in **AHLINets** iff (2) is a PB in **AHLNets** and (3) is a PB in **Sets** with the components of the **AHLINets** morphisms, where $m_0 : I_0 \to A \otimes P_0$ is induced by PB object P_0 in the commutating cube below (whose front's place components let the front commutate because of the PB in the net structure).

$$
\begin{tikzcd}
ANI_0 \arrow[r, "f_1"] \arrow[d, "f_2"'] \arrow[dr, phantom, "(1)"] & ANI_1 \arrow[d, "g_1"] \\
ANI_2 \arrow[r, "g_2"'] & ANI_3
\end{tikzcd}
\qquad
\begin{tikzcd}
I_0 \arrow[rr, "f_{1I}"] \arrow[dr, dashed, "m_0"] \arrow[dd, "f_{2I}"'] & & I_1 \arrow[dr, "m_1"] \arrow[dd, "g_{1I}" near start] & \\
& A_0 \otimes P_0 \arrow[rr, "f_{1A} \otimes f_{1P}" near start] \arrow[dd, "f_{2A} \otimes f_{2P}" near start] & & A_1 \otimes P_1 \arrow[dd, "g_{1A} \otimes g_{1P}"] \\
I_2 \arrow[rr, "g_{2I}"' near end] \arrow[dr, "m_2"'] & & I_3 \arrow[dr, "m_3"] & \\
& A_2 \otimes P_2 \arrow[rr, "g_{2A} \otimes g_{2P}"'] & & A_3 \otimes P_3
\end{tikzcd}
$$

$$
\begin{tikzcd}
AN_0 \arrow[r, "f'_1"] \arrow[d, "f'_2"'] \arrow[dr, phantom, "(2)"] & AN_1 \arrow[d, "g'_1"] \\
AN_2 \arrow[r, "g'_2"'] & AN_3
\end{tikzcd}
\quad
\begin{tikzcd}
I_0 \arrow[r, "f_{1I}"] \arrow[d, "f_{2I}"'] \arrow[dr, phantom, "(3)"] & I_1 \arrow[d, "g_{1I}"] \\
I_2 \arrow[r, "g_{2I}"'] & I_3
\end{tikzcd}
$$

Example. The pushouts in Fig. 7 on page 30 are pullbacks, too.

3.2. Transformation of AHLI Nets

This section is about rule-based transformation of AHLI nets. For this, we use the double pushout approach, which has also been used for *collective* AHL nets in [PER95]

[14] We require morphisms that are injective on the net structure in order to obtain componentwise pullbacks in **AHLNets**. See [EEPT06] for details.

and which has been investigated in the context of $\mathcal{M}$-adhesive systems in [EEPT06]. We are going to characterize the applicability of rules at some match by initial pushouts.

Definition 3.6 (AHLI Transformation Rules).
An AHLI transformation rule is a span of injective **AHLINets** morphisms

$$\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R).$$

Remark. AHLI Rule morphisms are not required to be marking-strict in order to obtain an $\mathcal{M}$-adhesive category (see Sect. 5 on page 41). This allows to arbitrarily change the marking of an AHLI net by applying rules with corresponding places and individual tokens in L and R.

Definition 3.7 (AHLI Transformation).
Given an AHLI transformation rule $\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ and an AHLI net ANI_1 with a AHLI net morphism $m : L \rightarrow ANI_1$, called the match, a direct AHLI net transformation $ANI_1 \overset{\varrho,m}{\Longrightarrow} ANI_2$ from ANI_1 to the AHLI net ANI_2 is given by the following double-pushout diagram (DPO) diagram in the category **AHLINets**:

$$\begin{tikzcd}
L \arrow[d, "m"'] \arrow[dr, phantom, "(PO_1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(PO_2)"] & R \arrow[d, "m^*"] \\
ANI_1 & ANI_0 \arrow[l] \arrow[r] & ANI_2
\end{tikzcd}$$

To be able to decide whether a rule is applicable at a certain match, we formulate a gluing condition for AHLI nets, such that there exists a pushout complement of the left rule morphisms and the match if (and only if) they fulfill the gluing condition. The correctness of the gluing condition is shown via a proof on initial pushouts over matches, according to [EEPT06].

Definition 3.8 (Gluing Condition in **AHLINets**).
Given AHLI nets K, L, and ANI and AHLI morphisms $l : K \rightarrow L$ and $f : L \rightarrow ANI$. We define the set of identification points[15]

$$IP = IP_P \cup IP_T \cup IP_I$$

with

- $IP_P = \{x \in P_L \mid \exists x' \neq x : f_P(x) = f_P(x')\}$,
- $IP_T = \{x \in T_L \mid \exists x' \neq x : f_T(x) = f_T(x')\}$,
- $IP_I = \{x \in I_L \mid \exists x' \neq x : f_I(x) = f_I(x')\}$,

the set of dangling points[16]

$$DP = DP_T \cup DP_I$$

with

[15] That is, all elements in L that are mapped non-injectively by f.
[16] That is, all places in L that would leave a dangling arc, if deleted.

- $DP_T = \{p \in P_L \mid \exists t \in T_{ANI} \setminus f_T(T_L) : f_P(p) \in ENV_P(t)\}$,
- $DP_I = \{p \in P_L \mid \exists i \in I_{ANI} \setminus f_I(I_L) : f_P(p) = \pi_P(m_{ANI}(i))\}$,

and the set of gluing points[17]

$$GP = l_P(P_K) \cup l_T(T_K) \cup l_I(I_K)$$

We say that l and f satisfy the gluing condition if $IP \cup DP \subseteq GP$. Given an AHLI rule $\varrho = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$, we say that ϱ and f satisfy the gluing condition iff l and f satisfy the gluing condition.

$$\begin{tikzcd} L \arrow[d, "f"'] & K \arrow[l, "l"'] \arrow[r, "r"] & R \\ ANI & & \end{tikzcd}$$

In order to construct the initial pushout for a match $f : L \to ANI$, we define the *boundary* over the match f, which is the minimal subnet containing all places, transitions, and individual tokens that must not be deleted by the application of a rule with left hand side L such that there exists a pushout complement.

Definition 3.9 (Boundary in **AHLINets**).
Given two AHLI nets

$$L = (\Sigma_L, P_L, T_L, pre_L, post_L, cond_L, type_L, A_L, I_L, m_L),$$
$$ANI = (\Sigma_{ANI}, P_{ANI}, T_{ANI}, pre_{ANI}, post_{ANI}, cond_{ANI}, type_{ANI}, A_{ANI}, I_{ANI}, m_{ANI})$$

and an AHLI morphism $f : L \to ANI$. The boundary of f is an AHLI net

$$B = (\Sigma_B, P_B, T_B, pre_B, post_B, cond_B, type_B, A_B, I_B, m_B)$$

with

- $\Sigma_B = \Sigma_L$,
- $P_B = DP_T \cup DP_I \cup IP_P \cup P_{IP_T} \cup P_{IP_I}$
- $P_{IP_T} = \{p \in P_L \mid \exists t \in IP_T : p \in ENV_P(t)\}$
- $P_{IP_I} = \{p \in P_L \mid \exists i \in IP_I : p = \pi_P(m_L(i))\}$
- $T_B = IP_T$
- $pre_B(t) = pre_L(t)$
- $post_B(t) = post_L(t)$

[17] That is, all elements in L that have a preimage in K.

- $cond_B(t) = cond_L(t)$
- $type_B(p) = type_L(p)$
- $A_B = A_L$
- $I_B = IP_I$
- $m_B(i) = m_L(i)$

together with $b : B \to L = (b_\Sigma, b_P, b_T, b_A, b_I)$ where $b_\Sigma = id_{\Sigma_B}$, $b_A = id_{A_B}$, and the remaining parts are inclusions.

Well-definedness.

$pre_B, post_B : T_B \to (T_{OP_B}(X_B) \otimes P_B)^\oplus$:
Let $t \in T_B$ and let $(term, p) \leq pre_B(t)$. Then there is $(term, p) \leq pre_L(t)$ which means that $term \in T_{OP_L}(X_L)$ and $p \in P_L$. Then from $\Sigma_B = \Sigma_L$ follows that $term \in T_{OP_B}(X_B)$. Furthermore there is $t \in IP_T$ which by the fact that $p \in ENV_P(t)$ means that $p \in P_{IP_T} \subseteq P_B$.
So pre_B is well-defined. The proof for $post_B$ works completely analogously.

$cond_B : T_B \to \mathcal{P}_{fin}(Eqns(\Sigma_B))$:
Let $t \in T_B$. Then we have

$$cond_B(t) = cond_L(t) \in \mathcal{P}_{fin}(Eqns(\Sigma_L)) = \mathcal{P}_{fin}(Eqns(\Sigma_B))$$

$type_B : P_B \to S_B$:
Let $p \in P_B$. Then we have

$$type_B(p) = type_L(p) \in S_L = S_B$$

$m_B : I_B \to A_B \otimes P_B$:
Let $i \in I_B$ and let $(a, p) = m_B(i)$. Then there is $(a, p) = m_L(i)$ which means that $a \in A_{type_L(p)} = A_{type_B(p)}$. The fact that $p \in P_B$ follows from the fact that $i \in IP_I$ and hence $p \in P_{IP_I} \subseteq P_B$.

inclusion $b : B \to L$:
We obtain an inclusion morphism $b : B \to L$ from the fact that pre_B,$post_B$,$cond_B$,$type_B$, and m_B are restrictions of the respective functions in L.

□

The following facts about the gluing condition and initial pushouts hold in all $\mathcal{M}$-adhesive categories $(\mathbf{AHLINets}, \mathcal{M})$ whose morphism class $\mathcal{M}$ of monomorphisms contains at least inclusions with identities for signature and algebra parts (for concrete instantiations see Sect. 5 on page 41). The next fact completes the construction of initial pushouts for matches and shows that the construction of the boundary in Def. 3.9 on the previous page complies to the categorical notion of boundaries in Def. A.1 on page 49.

Fact 3.10 (Initial Pushout in **AHLINets**).
Given a morphism $f : L \to ANI$ in **AHLINets**, the boundary B of f and the AHLI net

$$C = (\Sigma_C, P_C, T_C, pre_C, post_C, cond_C, type_C, A_C, I_C, m_C)$$

with

- $\Sigma_C = \Sigma_{ANI}$
- $P_C = (P_{ANI} \setminus f_P(P_L)) \cup f_P(b_P(P_B))$
- $T_C = (T_{ANI} \setminus f_T(T_L)) \cup f_T(b_T(T_B))$
- $pre_C(t) = pre_{ANI}(t)$
- $post_C(t) = post_{ANI}(t)$
- $cond_C(t) = cond_{ANI}(t)$
- $type_C(p) = type_{ANI}(p)$
- $A_C = A_{ANI}$
- $I_C = (I_{ANI} \setminus f_I(I_L)) \cup f_I(b_I(I_B))$
- $m_C(i) = m_{ANI}(i)$

Then diagram (1) is an initial pushout in **AHLINets**, where $g := f|_B$ and c is an inclusion with $c_\Sigma = id_{\Sigma_C}$ and $c_A = id_{A_C}$.

$$\begin{tikzcd}
B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(1)"] & L \arrow[d, "f"] \\
C \arrow[r, "c"'] & ANI
\end{tikzcd}$$

Proof. See section B.3. □

The following two facts show the correspondence between the gluing condition in **AHLINets** and the categorical gluing condition (see Def. A.2 on page 49), which is a necessary and sufficient condition for the (unique) existence of pushout complements in $\mathcal{M}$-adhesive categories.

Fact 3.11 (Characterization of Gluing Condition in **AHLINets**).
Let $l : K \to L$ and $f : L \to ANI$ be morphisms in **AHLINets** with $l \in \mathcal{M}$.
The morphisms l and f satisfy the gluing condition in **AHLINets** if and only if they satisfy the categorical gluing condition.

Proof. See section B.4. □

Fact 3.12 (Gluing Condition for AHLI Transformation).
Given an AHLI rule $\varrho = (L \xleftarrow{l} K \xrightarrow{r} R)$ and a match $f : L \to ANI$ into an AHLI net $ANI = (AN, I, m : I \to P_{AN})$. The rule ϱ is applicable on match f, i.e. there exists a (unique) pushout complement ANI_0 in the diagram below, iff ϱ and f satisfy the gluing condition in **AHLINets**.

$$\begin{tikzcd} L \arrow[d, "f"'] \arrow[dr, phantom, "(PO)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, dashed, "f'"] & R \\ ANI & ANI_0 \arrow[l, dashed] & \end{tikzcd}$$

Proof. By Fact 3.11 the rule ϱ and match f satisfy the gluing condition in **AHLINets** if and only if ϱ and f satisfy the categorical gluing condition, which by Fact A.3 is a sufficient and necessary condition for the existence of a (unique) pushout complement ANI_0. □

3.3. Correspondence of Transition Firing and Rules

Now that we can manipulate a net's marking with rules, we have a look at rules that simulate firing steps for a certain transition under a token selection. We show that firing of a transition corresponds to a canonical application of a special rule construction, called transition rules. With this correspondence we can easily show that token-injective AHLI morphisms preserve firing steps.

Definition 3.13 (AHLI Transition Rules).
Given an AHLI net

$$ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m)$$

we define the *transition rule* for a consistent transition assignment $(t, asg) \in CT_{ANI}$, enabled under the token selection $S = (M, m, N, n)$, as the rule

$$\varrho(t, S, asg) = (L_t \xleftarrow{l} K_t \xrightarrow{r} R_t)$$

with

- the common fixed AHL net part $AN_t = (\Sigma, P_t, \{t\}, pre_t, post_t, type_t, A)$, where $P_t = ENV_P(t), pre_t(t) = pre_(t), post_t(t) = post(t), type_t(p) = type(p)$,
- $L_t = (AN_t, M, m_t : M \to A \otimes P_t)$, with $m_t(x) = m(x)$,
- $K_t = (AN_t, \emptyset,\ \emptyset : \emptyset \to (A \times P_t))$,
- $R_t = (AN_t, N, n_t : N \to A \otimes P_t)$, with $n_t(x) = n(x)$,
- l, r being the obvious inclusions on the rule nets.

m_t and n_t are well-defined because t is enabled under S: The token selection condition implies that $\forall x \in M : m(x) \in PRE_A(t, asg)$ and due to the construction of AN_t we have $PRE_A(t, asg) \subseteq (A \otimes ENV_P(t)) = (A \otimes P_t)$. The argument for n_t works analogously.

Note that (t, asg) is enabled under S in L_t.

Remark. The structure of a transition rule depends only on the transition and the token selection, for which there may exist several enabled transition assignments. Therefore different consistent transition assignments may have the same correspondent transition rule. Nevertheless, we denote an AHLI transition rule as $\varrho(t, S, asg)$ rather than $\varrho(t, S)$ to remember the concrete assignment this rule is intended to simulate.

Example. Figure 7 shows an AHLI transition rule $\varrho(t, S, asg) = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$.

Definition 3.14 (Canonical DPO Transformation of AHLI Nets)**.**
We call a direct transformation $ANI_1 \overset{\varrho,f}{\Longrightarrow} ANI_2$ by rule $\varrho = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ with l, r being inclusions *canonical* if

- f_I is injective,
- f_Σ is injective on the set of variables of Σ_{ANI_1}[18]
- the morphisms in the span $(ANI_1 \leftarrow ANI_0 \rightarrow ANI_2)$ of the DPO transformation diagram below are inclusions, and
- $I_2 = I_0 \cup (I_R \setminus r(I_K))$.

$$\begin{tikzcd}
L \arrow[d, "f"'] \arrow[dr, phantom, "(PO_1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(PO_2)"] & R \arrow[d, "f^*"] \\
ANI_1 & ANI_0 \arrow[l] \arrow[r] & ANI_2
\end{tikzcd}$$

Remark. For each rule ϱ being applied to some AHLI net ANI at a token-injective match f, there exists a canonical transformation diagram, which is just the particular equivalent DPO diagram with "as less isomorphic changes as possible".

Because of the injectivity of the rule morphisms the lower span in the diagram is already injective as well. To construct the canonical transformation diagram we first regard the last condition on I_2, which demands that $(I_R \setminus r(I_K)) \cap I_0 = \emptyset$ because of pushout PO_2. For this, it is sufficient to replace the set of tokens in R that are created by the rule, i.e. $I_R \setminus r(I_K)$, such that it is disjoint to the tokens preserved by the rule, i.e. $(I_1 \setminus f(I_L)) \cup f(l(I_K))$.[19] With this we now can simply replace arbitrary ANI_0 and ANI_2, being some pushout complement object of PO_1 and pushout object of PO_2, with nets such that the lower span morphisms become inclusions.

[18]See the "Context Condition" in [EP97].

[19]This modification of the rule is passable since changing the tokens in R does not affect the firing behavior of NI_2 (up to the token selections) nor the applicability of the rule.

Example. The following diagram shows the two pushouts in **AHLINets** resulting from applying the AHL transition rule $\varrho(t, S, asg) = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$ for

$$asg = \{t_{11} \mapsto a_1, t_{12} \mapsto b_1, t_2 \mapsto a_2, t_3 \mapsto a_3\}$$

to the net ANI. All nets in this diagram have the same signature and algebra as the example net in Fig. 6 on page 20. Moreover, this transformation is canonical.

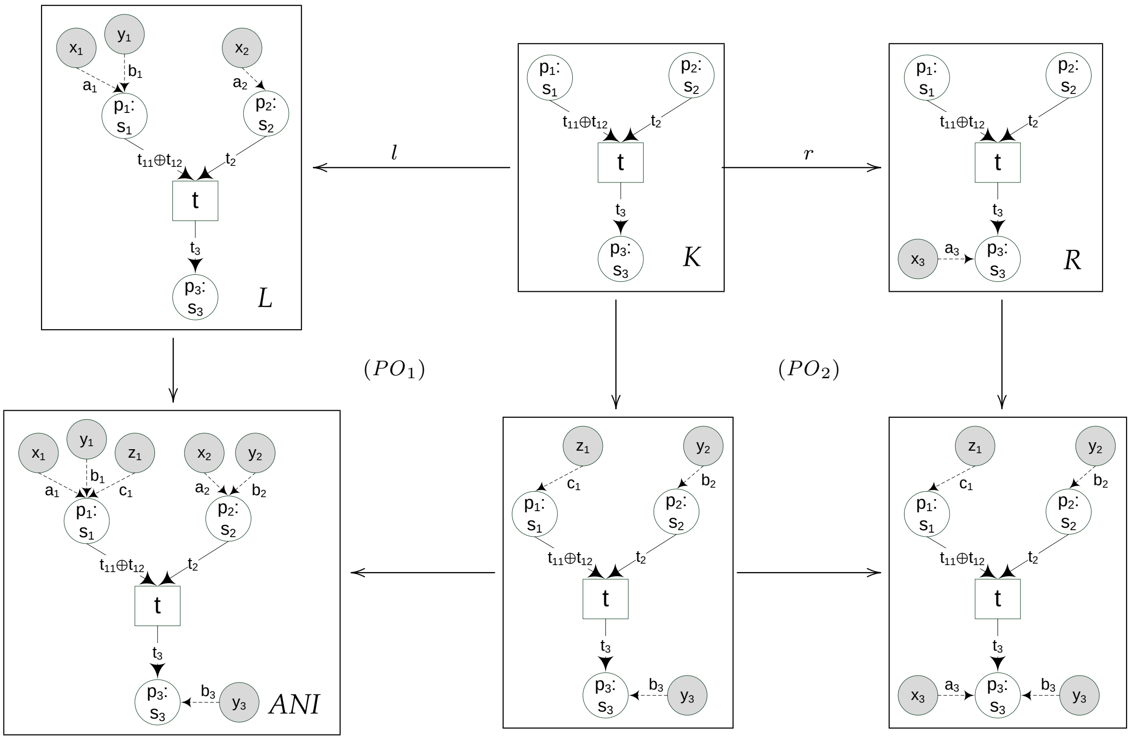

Figure 7: Canonical direct DPO-transformation in **AHLINets** simulating a firing step

Theorem 3.1 (Equivalence of Canonical Transformations and Firing of AHLI Nets).

1. *Each firing step $ANI \xrightarrow{t,asg} ANI'$ via token selection $S = (M, m, N, n)$ corresponds to a canonical direct transformation $ANI \xRightarrow{\varrho(t,S,asg),f} ANI'$ via the transition rule $\varrho(t, S, asg)$ matched by the inclusion match $f : L \to ANI$.*

2. *Each canonical direct transformation $ANI \xRightarrow{\varrho(t,S,asg),f} ANI_1$, via some transition rule $\varrho(t, S, asg) = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$ with $t \in T_{ANI}$ and token selection $S = (M, m, N, n)$, and token-injective match $f : L \to ANI$, implies the consistent transition assignment $(f_T(t), asg_f)$ being enabled in ANI under*

$$\left(f_I(M), \left((f_A \otimes f_P) \circ m \circ f_I^{-1}\right), N, \left((f_A^* \otimes f_P^*) \circ m_1 \circ f_I^{*-1}\right)\right)$$

with firing step $ANI \xrightarrow{f_T(t),asg_f} ANI_1$. *With* asg_f *we denote the translation of* asg *along* f, *i.e.* $asg_f = f_A \circ asg \circ f^{-1}_{\Sigma|Var(t)}$.

Proof.

1. Given the firing step $ANI \xrightarrow{t,asg} ANI'$ via $(M, m, N, n) = S$, the canonical direct transformation of the transition rule $\varrho(t, S, asg)$ is $ANI \xRightarrow{\varrho(t,asg),f} ANI_1$ as in the DPO diagram in Fig. 8. The match f, d, and d' are inclusions.

$$
\begin{tikzcd}
L = (AN_t, M, m_t) \arrow[d, "f"'] \arrow[dr, phantom, "(PO_1)"] & K = (AN_t, \emptyset, \emptyset) \arrow[l, hook', "l"'] \arrow[r, hook, "r"] \arrow[d] \arrow[dr, phantom, "(PO_2)"] & R = (AN_t, N, n_t) \arrow[d, "f^*"] \\
ANI = (AN, I, m) & ANI_0 = (AN, I_0, m_0) \arrow[l, hook', "d"] \arrow[r, hook, "d'"'] & ANI_1 = (AN, I_1, m_1)
\end{tikzcd}
$$

Figure 8: DPO diagram for canonical direct transformation of ANI with $\varrho(t, S, asg)$ in **AHLINets**

According to Fact 3.4 on page 22, we have $I_0 = I \setminus M$ and $I_1 = I_0 \cup (N \setminus \emptyset)$ as in the DPO diagram of the **Sets** components in Fig. 9. The firing step condition $(I\setminus M) \cap N = \emptyset$ grants us the last condition for canonical transformations $I_0 \cup (N \setminus r(\emptyset)) = I_1$.

$$
\begin{tikzcd}
M \arrow[d, "f_I"'] \arrow[dr, phantom, "(PO_1)"] & \emptyset \arrow[l, hook'] \arrow[rr, hook] \arrow[d] \arrow[drr, phantom, "(PO_2)"] & & N \arrow[d, "f_I^*"] \arrow[dr, "n_t"] & \\
I \arrow[drr, "m"'] & I_0 = (I \setminus M) \arrow[l, hook', "d_I"] \arrow[rr, hook, "d'_I"] \arrow[dr, dashed, "m_0 = m \circ d_I"] & & I_1 = (I \setminus M) \cup N \arrow[dr, dashed, "m_1"'] & A_t \otimes P_t \arrow[d, "f_A^* \otimes f_P^* = f_A \otimes f_P"] \\
& & A_{AN} \otimes P_{AN} \arrow[rr, Rightarrow, "id"'] & & A_{AN} \otimes P_{AN}
\end{tikzcd}
$$

Figure 9: DPO diagram in **Sets** for the token components of the transformation in Fig. 8

For m_1 as induced morphism for the pushout object I_1 follows that
$$m_1(x) = \begin{cases} m_0(x) = m(x) \text{ for } x \in I \setminus M \\ n_t(x) = n(x) \text{ for } x \in N \end{cases}$$
hence $I_1 = I', m_1 = m'$ and $ANI_1 = ANI'$, according to Def. 3.2 on page 20.

Remark. f_I being injective is not only sufficient for the existence of (PO_1), but also necessary, because $I_K = \emptyset$ (see Fact 3.12 on page 28).

2. Consider a canonical direct transformation $ANI \xRightarrow{\varrho(t,S,asg),f} ANI_1$ with

$$ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m)$$

as in the DPO diagrams in Figs. 8 and 9. $(f_T(t), asg_f) \in CT_{ANI}$ is enabled under

$$\big(f_I(M), \big((f_A \otimes f_P) \circ m \circ f_I^{-1}\big), N, \big((f_A^* \otimes f_P^*) \circ m_1 \circ f_I^{*-1}\big)\big)$$

if

1. $f_I(M) \subseteq I$,
2. $(f_A \otimes f_P) \circ m \circ f_I^{-1} : N \to A \otimes P$,
3. $(I \setminus f_I(M)) \cap N = \emptyset$,
4. $\sum_{i \in f_I(M)} \big((f_A \otimes f_P) \circ m \circ f_I^{-1}(i)\big) = pre_A(f_T(t), asg_f)$
5. $\sum_{i \in N} \big((f_A^* \otimes f_P^*) \circ m_1 \circ f_I^{*-1}(i)\big) = post_A(f_T(t), asg_f)$
6. leading to the follower marking $I' = (I \setminus M) \cup N$ with $m'(x) = \begin{cases} m_0(x) = m(x) \text{ for } x \in I \setminus M \\ m_1(x) = n(x) \text{ for } x \in N \end{cases}$

By construction of the transition rule $\varrho(t, S, asg)$ and its application to ANI we have 1., 2., and for 6. that $I' = I_1, m' = m_1$. The canonical transformation property grants us 3. It remains to show 4. and 5., i.e. that $f_I(M)$ and N represent the correct numbers of value/place pairs in the environment of $f_T(t)$ to enable it when evaluated with asg_f:

$$
\begin{aligned}
&\sum_{i \in f_I(M)} (f_A \otimes f_P) \circ m \circ f_I^{-1}(i) \\
&= \sum_{i \in M} (f_A \otimes f_P) \circ m(i) && (f_I \text{ inj.}) \\
&= (f_A \otimes f_P)^{\oplus} \sum_{i \in M} m_t(i) && (\forall i \in M : m_t(i) = m(i), \text{due to def. } p(t, S, asg)) \\
&= (f_A \otimes f_P)^{\oplus} \circ (\overline{asg} \times id_P)^{\oplus} \circ pre_{AN_t}(t) && ((t, asg) \text{ enabled in } L) \\
&= (\overline{asg_f} \times id_P)^{\oplus} \circ (f_\Sigma^{\#} \otimes f_P)^{\oplus} \circ pre_{AN_t}(t) && (\text{Lemma 3.2}) \\
&= (\overline{asg_f} \times id_P)^{\oplus} \circ pre \circ f_T(t) && (f \text{ AHLI morph.}) \\
&= pre_A(f_T(t), asg_f) && (\text{def. } pre_A)
\end{aligned}
$$

and analogously,

$$
\begin{aligned}
&\sum_{i\in N}(f_A^*\otimes f_P^*)\circ m_1\circ f_I^{*-1}(i) \\
&= \sum_{i\in N}(f_A^*\otimes f_P^*)\circ m_1(i) && (f_I^*(N)=N) \\
&= (f_A^*\otimes f_P^*)^{\oplus}\sum_{i\in M} n_t(i) && (\forall i\in N: n_t(i)=m_1(i),\text{due to constr. } m_1) \\
&= (f_A^*\otimes f_P^*)^{\oplus}\circ(\overline{asg}\times id_P)^{\oplus}\circ pre_{AN_t}(t) && ((t,asg)\text{ enabled in } L) \\
&= (\overline{asg_f}\times id_P)^{\oplus}\circ(f_\Sigma^{\#}\otimes f_P)^{\oplus}\circ post_{AN_t}(t) && (\text{Lemma 3.2}) \\
&= (\overline{asg_f}\times id_P)^{\oplus}\circ post\circ f_T(t) && (f\text{ AHLI morph.}) \\
&= post_A(f_T(t),asg_f) && (\text{def. } post_A)
\end{aligned}
$$

□

Lemma 3.2 (Translated Assignments).
Given two AHLI nets

$$ANI_i=(\Sigma_i=(S_i,OP_i,X_i),P_i,T_i,pre_i,post_i,cond_i,type_i,A_i,I_i,m_i), i\in\{1,2\}$$

a transition assignment $(t,asg: X_1\to A_1)\in CT_{ANI_1}$*, and an AHLI net morphism* $f=(f_\Sigma,f_P,f_T,f_A,f_I): ANI_1\to ANI_2$ *with* f_Σ *injective on the variables of* $Var(t)$*, it holds for all terms* $term\in T_{OP_1}(Var(t))$ *that*

$$\overline{asg_f}\circ f_\Sigma^{\#}(term)=f_A\circ\overline{asg}(term),\quad \text{where } asg_f=f_A\circ asg\circ f^{-1}_{\Sigma|Var(t)}: Var(f_T(t))\to A_2$$

Proof by structural induction over all Σ_1 *terms over variables of* $Var(t)$. [20] First, note that because (f_Σ,f_A) is a generalized algebra homomorphism we have for all constants c and operations op in OP_1

$$f_A(c_{A_1})\overset{f_A\text{ homomorph.}}{=}c_{V_{f_\Sigma}(A_2)}\overset{\text{def. }V_{f_\Sigma}}{=}(f_\Sigma(c))_{A_2} \tag{1}$$

$$f_A\circ op_{A_1}\overset{f_A\text{ homomorph.}}{=}op_{V_{f_\Sigma}(A_2)}\circ f_A\overset{\text{def. }V_{f_\Sigma}}{=}(f_\Sigma(op))_{A_2}\circ f_A \tag{2}$$

Case 1: $t=x\in Var(t)$
$f_A\circ\overline{asg}(x)=\overline{f_A\circ asg}(x)=\overline{f_A\circ asg\circ f^{-1}_{\Sigma|Var(t)}}\circ f_\Sigma^{\#}(x)$
because $f_{\Sigma|Var(t)}: Var(t)\to Var(f_T(t))$ is surjective due to f being a net morphism and hence f_Σ bijective on variables in $Var(t)$.

Case 2: $t=(c:\to s)\in T_{OP_1}$
$f_A\circ\overline{asg}(c)=f_A(c_{A_1})\overset{(1)}{=}(f_\Sigma(c))_{A_2}=(f_\Sigma^{\#}(c))_{A_2}=\overline{asg_f}\circ f_\Sigma^{\#}(c)$
The last equality holds because a constant c would be evaluated by the extension of just any variable assignment to c_{A_2}.

[20] A categorical proof using free constructions can be found in [EP97].

Case 3: $t = op(t_1, \ldots, t_n) \in T_{OP_1}(Var(t))$ with $t_1, \ldots, t_n$ satisfying the property to be proven.

$$\begin{aligned}
f_A \circ \overline{asg}(op(t_1,\ldots,t_n)) &= f_A \circ op_{A_1}(\overline{asg}(t_1),\ldots,\overline{asg}(t_n))\\
&\overset{(2)}{=}(f_\Sigma(op))_{V_{f_\Sigma}(A_2)}(f_A \circ \overline{asg}(t_1),\ldots,f_A \circ \overline{asg}(t_n))\\
&\overset{ind.assumpt.}{=}(f_\Sigma(op))_{V_{f_\Sigma}(A_2)}(\overline{asg_f} \circ f_\Sigma^{\#}(t_1),\ldots,\overline{asg_f} \circ f_\Sigma^{\#}(t_n))\\
&=\overline{asg_f}\left((f_\Sigma(op))\,(f_\Sigma^{\#}(t_1),\ldots,f_\Sigma^{\#}(t_n))\right)\\
&=\overline{asg_f} \circ f_\Sigma^{\#}\,(op(t_1,\ldots,t_n))
\end{aligned}$$

□

The second item of the previous theorem covers all possible canonical direct transformations by any transition rule $\varrho(t, S, asg)$ in an arbitrary net ANI, without aasuming that $t \in ANI$ or $M \subseteq I_{NI}$. For the special case that a transition rule $\varrho(t, S, asg)$ is applied on an inclusion match, the second item reduces to the following corollary, which is more similar to the theorem's first item.

Corollary 3.3 (Equivalence of Canonical Transformations and Firing of AHLI Nets)**.** *Given a canonical transformation* $ANI \xRightarrow{\varrho(t,S,asg),f} ANI_1$ *such that the match* $f : L \to ANI$ *is an inclusion, then the consistent token assignment* (t, asg) *is enabled in* ANI *under* S *with firing step* $ANI \xrightarrow{t,asg} ANI_1$.

This follows directly from 2. of Theorem 3.1.

Theorem 3.4 (Token-injective AHLI Net Morphisms preserve Firing Steps)**.** *For each AHLI net morphism* $f : ANI_1 \to ANI_2$*, such that* f_I *is injective and* f_Σ *is injective on all sets* $Var(t)$ *for all* $t \in T_1$[21]*, and each firing step* $ANI_1 \xrightarrow{t,asg} ANI'_1$ *there exists a firing step* $ANI_2 \xrightarrow{f_T(t),asg_f} ANI'_2$ *and a AHLI net morphism* $f' : ANI'_1 \to ANI'_2$ *(depicted in diagram (1) below).*

$$\begin{tikzcd}
ANI_1 \arrow[r, tail, "t{,}asg"] \arrow[d, "f"'] \arrow[dr, phantom, "(1)"] & ANI'_1 \arrow[d, "f'"] \\
ANI_2 \arrow[r, tail, "f_T(t){,}asg_f"'] & ANI'_2
\end{tikzcd}$$

Proof. Given $f : ANI_1 \to ANI_2$ and $ANI_1 \xrightarrow{t,asg} ANI'_1$ via some S as above, we have by the first part of Theorem 3.1 on page 30 the canonical direct transformation with $\varrho(t, S, asg) = (L \xleftarrow{l} K \xrightarrow{r} R)$ given by the pushouts (1) and (2) in Fig. 10 on the facing page.

Note that $f : ANI_1 \to ANI_2$ satisfies the gluing condition w.r.t l', because $l'_P = id_{P_1}, l'_T = id_{T_1}$ and $IP_I = \emptyset$ due to injectivity of f_I. This allows to construct the

[21] See the "Context Condition" in [EP97].

$$\begin{tikzcd}
L \arrow[d, "inc_L"'] \arrow[dr, phantom, "(1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d] \arrow[dr, phantom, "(2)"] & R \arrow[d, "inc_R"] \\
ANI_1 \arrow[d, "f"'] \arrow[dr, phantom, "(3)"] & ANI_{D1} \arrow[l, "l'"'] \arrow[r, "r'"] \arrow[d] \arrow[dr, phantom, "(4)"] & ANI'_1 \arrow[d, "f'"] \\
ANI_2 & ANI_{D2} \arrow[l] \arrow[r] & ANI'_2
\end{tikzcd}$$

Figure 10: Double pushout diagrams for canonical direct transformation of ANI_1 and ANI_2 with $\varrho(t, S, asg)$

canonical transformation with extended rule $ANI_1 \overset{l'}{\leftarrow} ANI_{D1} \overset{r'}{\rightarrow} ANI'_1$ along pushouts (3) and (4). Hence, also $(1+3)$ and $(2+4)$ are pushouts, defining a canonical direct transformation via $\varrho(t, S, asg)$. The second part of Theorem 3.1 on page 30 implies $ANI_2 \xrightarrow{f_T(t), asg_f} ANI'_2$. □

Remark (Firing-preserving diagrams for (t, asg) correspond to an extension diagram for $\varrho(t, asg)$). The diagram in Fig. 10 corresponds to an extension diagram for rule $\varrho(t, asg)$ and f (as depicted below) because $(1)-(4)$ are pushouts.

$$\begin{tikzcd}
ANI_1 \arrow[r, Rightarrow, "\varrho(t,asg),inc"] \arrow[d, "f"'] & ANI'_1 \arrow[d, "f'"] \\
ANI_2 \arrow[r, Rightarrow, "\varrho(t,asg),f\circ inc"] & ANI'_2
\end{tikzcd}$$

4. Functors for Individual Net Classes

PTI and AHLI nets are still very close to their *collective* pendants. In this section, we express this vicinity with functors. In addition we define a flattening functor from AHLI to PTI nets, similar to the flattening of AHL nets, that preserves enabling and firing and finally show a compatibility result for these functors.

Definition and Fact 4.1 ($Coll_{PT} : \mathbf{PTINets}_I \to \mathbf{PTSys}$).
The following construction flattens a PTI net to a P/T net with collective marking and forgets the individual token elements. We can translate nets with a finite number of tokens on each place, only:

$$Coll_{PT}(P, T, pre, post, I, m) = \left(P, T, pre, post, M = \sum_{i \in I} m(i) \in P^{\oplus}\right),$$

if $\forall p \in P : |m^{-1}(p)| \in \mathbb{N}$.

Now, we extend the construction to a functor $Coll_{PT} : \mathbf{PTINets}_I \to \mathbf{PTSys}$. **PTSys** is the category of P/T nets with a marking $M \in P^{\oplus}$, where $P^{\oplus}$ is the commutative monoid over the set of places. Morphisms in **PTSys** are pairs

$$(f_P : P_1 \to P_2, f_T : T_1 \to T_2) : N_1 \to N_2$$

that are compatible to the nets' *pre* and *post* mappings (just as PTI morphisms) and preserve markings placewise, i.e. $\forall p \in P_1 : M_{N_1}(p) \leq M_{N_1}(f_P(p))$.

Because markings of nets in **PTSys** can only describe finitely many tokens on each place of the net, we define as domain of the functor the category $\mathbf{PTINets}_I$ of PTI nets with finitely many tokens on each place (such that $\forall p \in P : |m^{-1}(p)| \in \mathbb{N}$) and token-injective PTI morphisms. For the morphisms, we just have to forget the individual token component by defining

$$Coll_{PT}(f : NI_1 \to NI_2) = (f_P, f_T) : Coll_{PT}(NI_1) \to Coll_{PT}(NI_2).$$

Proof. Obviously, $Coll_{PT}(f)$ is well-defined on the net structure parts of $Coll_{PT}(NI_1)$ and $Coll_{PT}(NI_2)$. Furthermore, it holds that

$$\forall p \in P_1 : M_1(p) = |m_1^{-1}(p)| \leq |m_2^{-1}(f_P(p))| = M_2(f_P(p)).$$

The inequality is valid because each token x on place p implies a unique image $f_I(x)$ on $f_P(p)$ due to the PTI morphism properties and because f_I being injective does not merge token images in I_2.

The compositionality follows directly from the componentwise composition of PTI net morphisms. □

Theorem 4.1 ($Coll_{PT}$ preserves Enabling and Firing).
Given a PTI net

$$NI = (P, T, pre, post, I, m) \in \mathbf{PTINets}_I,$$

each valid token selection (M, m, N, n) *enabling a firing* $NI \xrightarrow{t} NI'$ *implies the firing* $Coll_{PT}(NI) \xrightarrow{t} Coll_{PT}(NI')$.

Proof.

1. t is enabled in $Coll_{PT}(NI)$:

$$pre(t) \overset{t \text{ enabled}}{=} \sum_{i \in M} m(i) \overset{M \subseteq I}{\leq} \sum_{i \in I} m(i) \overset{\text{def. } Coll_{PT}}{=} M_{Coll_{PT}(NI)}$$

2. Firing of t results in $Coll_{PT}(NI')$: Obviously, for the firing step $Coll_{PT}(NI) \xrightarrow{t} \widetilde{PN}$ the net structure parts of $\widetilde{PN}$ are the same as of $Coll_{PT}(NI')$. It remains to show the equality of their markings. We have

$$I_{NI'} = (I \setminus M) \uplus N, \quad m' : I_{NI'} \to P \text{ with } m'(x) = \begin{cases} m(x), & x \in I \setminus M \\ n(x), & x \in N \end{cases}$$

which we use to show that

$$
\begin{aligned}
& M_{\widetilde{PN}} \\
&= M_{Coll_{PT}(NI)} \ominus pre(t) \oplus post(t) \\
&= \sum_{i \in I} m(i) \ominus \sum_{i \in M} m(i) \oplus \sum_{i \in N} n(i) && \text{(def. } Coll_{PT}, t \text{ enabled)} \\
&= \sum_{i \in I \setminus M} m(i) \oplus \sum_{i \in N} n(i) \\
&= \sum_{i \in (I \setminus M) \uplus N} m'(i) && \text{(def. } m'\text{)} \\
&= M_{Coll_{PT}(NI')} && \text{(defs. } I_{NI'}, Coll_{PT}\text{)}
\end{aligned}
$$

□

Remark. Because the only condition for t to be enabled in $Coll_{PT}(NI)$ is that $\forall p \in P : pre(t)(p) \leq \left(\sum_{i \in I} m(i) \right)(p)$, there are possibly many different valid token selections corresponding to the same firing step of t in $Coll_{PT}(NI)$, depending only on isomorphic N and n.

Definition and Fact 4.2 ($Coll_{AHL} : \mathbf{AHLINets}_I \to \mathbf{AHLSys}$)**.**
The following construction flattens a AHLI net to a AHL net with collective marking and forgets the individual token elements. We can translate nets with a finite number of each value/place pairs, only:

$$
Coll_{AHL}(AN, I, m) = \left(AN, M = \sum_{i \in I} m(i) \in (A \otimes P)^{\oplus} \right),
$$

if $\forall (a,p) \in A \otimes P : |m^{-1}(a,p)| \in \mathbb{N}$.

Now, we extend the construction to a functor $Coll_{AHL} : \mathbf{AHLINets}_I \to \mathbf{AHLSys}$. **AHLSys** is the category of AHL nets with a marking $M \in (A \otimes P)^{\oplus}$, where $(A \otimes P)^{\oplus}$ is the commutative monoid over pairs of values from the net's algebra and the net's places of compatible type. Morphisms in **AHLSys** are tuples

$$
(f_\Sigma : \Sigma_1 \to \Sigma_2, f_P : P_1 \to P_2, f_T : T_1 \to T_2, f_A : A_1 \to A_2) : AN_1 \to AN_2
$$

that comply to all compatibility properties of AHLI net morphisms (of course, except the one regarding the individual token component) and that preserve markings place/valuewise, i.e.

$$
\forall (a,p) \in A_1 \otimes P_1 : M_{AN_1}(a,p) \leq M_{AN_2}(f_A \otimes f_P(a,p)).
$$

Because markings of nets in **AHLSys** can only describe finitely many tokens on each place of the net, we define as domain of the functor the category $\mathbf{AHLINets}_I$ of AHLI nets with finitely many occurrences of each value/place pair, i.e.

$$
\forall (a,p) \in A \otimes P : |m^{-1}(a,p)| \in \mathbb{N},
$$

and AHLI morphisms that are injective on tokens and variables[22]. For the morphisms, we just have to forget the individual token component by defining

$$Coll_{AHL}(f : ANI_1 \to ANI_2) = (f_\Sigma, f_P, f_T, f_A) : Coll_{AHL}(ANI_1) \to Coll_{AHL}(ANI_2).$$

Proof. Obviously, $Coll_{AHL}(f)$ is well-defined on the net structure parts of $Coll_{AHL}(ANI_1)$ and $Coll_{AHL}(ANI_2)$. All properties for AHL net morphisms are already valid for the correspondent components of AHLI net morphisms.

The compositionality follows directly from the componentwise composition of AHLI net morphisms. □

Theorem 4.2 ($Coll_{AHL}$ preserves Enabling and Firing)**.**
Given an AHLI net

$$ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m) \in \mathbf{AHLINets}_I,$$

each valid token selection (M, m, N, n) *with firing step* $ANI \overset{(t,asg)}{\rightarrowtail} ANI'$ *implies the firing* $Coll_{AHL}(ANI) \overset{(t,asg)}{\rightarrowtail} Coll_{AHL}(ANI')$.

Proof.
1. (t, asg) is enabled in $Coll_{AHL}(ANI)$:

$$pre_A(t, asg) \overset{(t,asg)\text{ enabled}}{=} \sum_{i \in M} m(i) \overset{M \subseteq I}{\leq} \sum_{i \in I} m(i) \overset{\text{def. } Coll_{AHL}}{=} M_{Coll_{AHL}(ANI)}$$

2. Firing of (t, asg) results in $Coll_{AHL}(ANI')$: Obviously, for the firing step $Coll_{AHL}(ANI) \overset{(t,asg)}{\rightarrowtail} \widetilde{ANI}$ the net structure parts of $\widetilde{ANI}$ are the same as of $Coll_{AHL}(ANI')$. It remains to show the equality of their markings. We have

$$I_{ANI'} = (I \setminus M) \uplus N, \quad m' : I_{ANI'} \to A \otimes P \text{ with } m'(x) = \begin{cases} m(x), & x \in I \setminus M \\ n(x), & x \in N \end{cases}$$

which we use to show that

$$\begin{aligned}
& M_{\widetilde{ANI}} \\
&= M_{Coll_{AHL}(ANI)} \ominus pre_A(t, asg) \oplus post_A(t, asg) \\
&= \sum_{i \in I} m(i) \ominus \sum_{i \in M} m(i) \oplus \sum_{i \in N} n(i) && \text{(def. } Coll_{AHL}, t \text{ enabled)} \\
&= \sum_{i \in I \setminus M} m(i) \oplus \sum_{i \in N} n(i) \\
&= \sum_{i \in (I \setminus M) \uplus N} m'(i) && \text{(def. } m'\text{)} \\
&= M_{Coll_{AHL}(ANI')} && \text{(defs. } I_{ANI'}, Coll_{AHL}\text{)}
\end{aligned}$$

□

[22] We need injectivity on variables of a signature for the flattening of AHLI morphisms.

Definition and Fact 4.3 ($Flat_I : \mathbf{AHLINets_I} \to \mathbf{PTINets_I}$)**.**
The following construction flattens an AHLI net to a PTI net:

$$Flat_I(AN, I, m) = (CP, CT, pre_A, post_A, I, m)$$

Note that $(CP, CT, pre_A, post_A) = Flat(AN)$ and $m : I \to (A \otimes P) = CP$.
To extend the construction to a functor $Flat_I : \mathbf{AHLINets_I} \to \mathbf{PTINets_I}$, we define for each

$$f = (f_\Sigma, f_P, f_T, f_A, f_I) : ANI_1 \to ANI_2$$

the flattening

$$Flat_I(f) = (f_A \otimes f_P, f_T \times \left(f_A \circ \_ \circ f^{-1}_{\Sigma|Var(t)}\right), f_I) : Flat_I(ANI_1) \to Flat_I(ANI_2).$$

Proof. To prove that

$$Flat_I(f : ANI_1 \to ANI_2) : Flat_I(ANI_1) \to Flat_I(ANI_2)$$

is a valid PTI net morphism, we have to show the following equalities:

$$f_I \circ m_1 = m_2 \circ (f_A \otimes f_P)$$

and

$$(f_A \otimes f_P)^\oplus \circ pre_{A1} = pre_{A2} \circ \left(f_T \times \left(f_A \circ \_ \circ f^{-1}_{\Sigma|Var(t)}\right)\right),$$

analogously for $post_A$. The first one follows directly from f being a AHLI morphism.
For the second one we have for all $(t, asg) \in CT_1$ that

$$\begin{aligned}
& pre_{A2} \circ \left(f_T \times \left(f_A \circ _ \circ f^{-1}_{\Sigma|Var(t)}\right)\right)(t, asg) \\
&= pre_{A2}\left(f_T(t), f_A \circ asg \circ f^{-1}_{\Sigma|Var(t)}\right) \\
&= pre_{A2}\left(f_T(t), asg_f\right) && \left(\text{abbrev. } asg_f = f_A \circ asg \circ f^{-1}_{\Sigma|Var(t)}\right) \\
&= (\overline{asg_f}, id_{P_2})^\oplus \circ pre_2 \circ f_T(t) && (\text{def. } pre_{A2}) \\
&= (\overline{asg_f}, id_{P_2})^\oplus \circ (f^\#_\Sigma \otimes f_P)^\oplus \circ pre_1(t) && (f \text{ AHLI morph.}) \\
&= \sum_{i=1}^n \left(\overline{asg_f} \circ f^\#_\Sigma(term_i), f_P(p_i)\right) && \text{for } \sum_{i=1}^n (term_i, p_i) = pre_1(t) \\
&= \sum_{i=1}^n (f_A \circ \overline{asg}(term_i), f_P(p_i)) && (\text{Lemma 3.2}) \\
&= (f_A \otimes f_P)^\oplus \circ (\overline{asg}, id_{P_1})^\oplus \circ pre_1(t) \\
&= (f_A \otimes f_P)^\oplus \circ pre_{A1}(t, asg) && (\text{def. } pre_{A1})
\end{aligned}$$

and analogously for $post_A$. The compositionality follows directly from the componentwise composition of AHLI net morphisms. □

Theorem 4.3 ($Flat_I$ preserves and reflects Enabling and Firing)**.**
Given an AHLI net $ANI = (AN, I, m)$ with $Flat_I(AN, I, m) = (CN, I, m)$ and a token selection $S = (M, m, N, n)$, the following holds:

1. *$(t, asg) \in CT$ is enabled under S in ANI, iff (t, asg) is enabled under S in $Flat_I(ANI)$.*

2. *$ANI \xrightarrow{(t,asg)} (AN, I', m')$ via S $\quad\Leftrightarrow\quad$ $(CN, I, m) \xrightarrow{(t,asg)} (CN, I', m')$ via S.*

Proof.

1. $(t, asg) \in CT$ is enabled under S in (AN, I, m)
$\Leftrightarrow \sum\limits_{i \in M} m(i) = pre_A(t, asg) \wedge \sum\limits_{i \in N} n(i) = post_A(t, asg)$
$\Leftrightarrow (t, asg) \in CT$ is enabled under S in $Flat_I(AN, I, m) = (CP, CT, pre_A, post_A, I, m)$

2. $(AN, I, m) \xrightarrow{(t,asg)} (AN, I', m')$ via S
$\Leftrightarrow (t, asg)$ enabled under S in both ANI and $Flat_I(ANI)$,

and $I' = (I \setminus M) \uplus N, \quad m'(x) = \begin{cases} m(x), & x \in I \setminus M \\ n(x), & x \in N \end{cases}$

$\Leftrightarrow Flat_I(AN, I, m) = (CN, I, m) \xrightarrow{(t,asg)} (CN, I', m') = Flat_I(AN, I', m')$ via S $\square$

Theorem 4.4 (Compatibility of Collection and Flattening Functors)**.**
The previously defined functors are compatible, i.e. $Coll_{PT} \circ Flat_I = Flat \circ Coll_{AHL}$.

$$\begin{tikzcd} \mathbf{AHLINets_I} \arrow[r, "Flat_I"] \arrow[d, "Coll_{AHL}"'] \arrow[dr, phantom, "="] & \mathbf{PTINets_I} \arrow[d, "Coll_{PT}"] \\ \mathbf{AHLSys} \arrow[r, "Flat"] & \mathbf{PTSys} \end{tikzcd}$$

Proof. Given an AHLI net $ANI = (\Sigma, P, T, pre, post, cond, type, A, I, m) \in \mathbf{AHLINets}_I$, we have

$$\begin{aligned} & Coll_{PT} \circ Flat_I(ANI) \\ & = Coll_{PT}(CP, CT, pre_A, post_A, I, m) && \text{(def. } Flat_I) \\ & = \left(CP, CT, pre_A, post_A, M = \sum_{i \in I} m(i) \in (A \otimes P)^{\oplus}\right) && \text{(def. } Coll_{PT}) \\ & = Flat\left(\Sigma, P, T, pre, post, cond, type, A, M = \sum_{i \in I} m(i)\right) && \text{(def. } Flat) \\ & = Flat \circ Coll_{AHL}(ANI) && \text{(def. } Coll_{AHL}) \end{aligned}$$

and for some AHL net morphism $f = (f_\Sigma, f_P, f_T, f_A, f_I) : ANI_1 \to ANI_2$ with injective

f_I we have

$$
\begin{aligned}
& Coll_{PT} \circ Flat_I(f) \\
& = Coll_{PT}(f_A \otimes f_P, f_T \times \left(f_A \circ \_ \circ f^{-1}_{\Sigma|Var(t)}\right), f_I) && \text{(def. } Flat_I) \\
& = (f_A \otimes f_P, f_T \times \left(f_A \circ \_ \circ f^{-1}_{\Sigma|Var(t)}\right)) && \text{(def. } Coll_{PT}) \\
& = Flat(f_\Sigma, f_P, f_T, f_A) && \text{(def. } Flat) \\
& = Flat \circ Coll_{AHL}(f) && \text{(def. } Coll_{AHL})
\end{aligned}
$$

□

5. Nets with Individual Tokens are $\mathcal{M}$-adhesive Categories

[Pra08] shows how to construct a marking category for *collective* Petri nets, which can be used to prove the properties of $\mathcal{M}$-adhesive categories, i.e. weak adhesive high-level replacement systems in the sense of [EEPT06], for marked Petri nets. In this section we present a similar construction for individual markings and show some instantiations of $\mathcal{M}$-adhesive categories for PTI and AHLI nets.

Definition 5.1 (Individual Petri Systems)**.**
Given a category **Nets** of nets, an *individual system* $IS = (N, I, m)$ is given by a net $N \in \textbf{Nets}$, a set of individuals I, and a function $m : I \to M(N)$, where $M : \textbf{Nets} \to \textbf{Sets}$ is a functor assigning a marking set to each net N.

For systems $IS_1 = (N_1, I_1, m_1)$ and $IS_2 = (N_2, I_2, m_2)$, an individual system morphism $f_{IS} = (f_N, f_I) : IS_1 \to IS_2$ consists of a net morphisms $f_N : N_1 \to N_2$ and a function on the individuals $f_I : I_1 \to I_2$ such that $M(f_N) \circ m_1 = m_2 \circ f_I$ as depicted below.

$$
\begin{tikzcd}
I_1 \arrow[r, "m_1"] \arrow[d, "f_I"'] \arrow[dr, phantom, "="] & M(N_1) \arrow[d, "M(f_N)"] \\
I_2 \arrow[r, "m_2"] & M(N_2)
\end{tikzcd}
$$

All individual systems and system morphisms constitute the category $\textbf{ISystems}(\textbf{Nets}, M)$.

Theorem 5.1 ($\textbf{ISystems}(\textbf{Nets}, M)$ is $\mathcal{M}$-adhesive)**.**
Given an $\mathcal{M}$-adhesive category $(\textbf{Nets}, \mathcal{M})$ of nets with a marking set functor $M : \textbf{Nets} \to \textbf{Sets}$ that preserves pullbacks along $\mathcal{M}$-morphisms, then the category

$$(\textbf{ISystems}(\textbf{Nets}, M), \mathcal{M} \times \mathcal{M}_{inj})$$

of individual systems over these nets is $\mathcal{M}$-adhesive as well, where $\mathcal{M}_{inj}$ is the class of all injective functions.

Proof. Consider the comma category $\mathbf{C} = ComCat(ID_{\textbf{Sets}}, M, \{m\})$, with objects $(I, N, op_m : I \to M(N))$ and morphisms $f_{\mathbf{C}} = (f_I, f_N) \in Mor_{\textbf{Sets}} \times Mor_{\textbf{Nets}}$. Obviously, $\mathbf{C}$ is isomorphic to $\textbf{ISystems}(\textbf{Nets}, M)$.

By applying Thm. 1.4 (Construction of [Weak] Adhesive HLR Categories) from [PEL08], we can conclude that $(\mathbf{C}, \mathcal{M} \times \mathcal{M}_{inj})$ is $\mathcal{M}$-adhesive because

- $(\mathbf{Nets}, \mathcal{M})$ and $(\mathbf{Sets}, \mathcal{M}_{inj})$ are $\mathcal{M}$-adhesive categories,
- the functor $ID_{\mathbf{Sets}}$ preserves pushouts along injective functions, and
- the functor $M : \mathbf{Nets} \to \mathbf{Sets}$ preserves pullbacks along $\mathcal{M}$-morphisms.

Hence, $(\mathbf{ISystems}(\mathbf{Nets}, M), \mathcal{M} \times \mathcal{M}_{inj})$ is $\mathcal{M}$-adhesive as well. □

Theorem 5.2 ($\mathbf{PTINets}$ is $\mathcal{M}$-adhesive)**.**
The category $(\mathbf{PTINets}, \mathcal{M}_{inj})$ *is an* $\mathcal{M}$*-adhesive category where*

$$\mathcal{M}_{inj} = \{f \in Mor_{\mathbf{PTINets}} | f_P, f_T, f_I \textit{ injective}\}$$

Proof. We already know from [Pra08] that $(\mathbf{PTNets}, \mathcal{M}')$ is $\mathcal{M}$-adhesive with $\mathcal{M}'$ being the class of all injective Petri net morphisms. Given the functor $M : \mathbf{PTNets} \to \mathbf{Sets}$ with $M(P, T, pre, post) = P$, we have $\mathbf{PTINets} \sim \mathbf{ISystems}(\mathbf{PTNets}, M)$.

M preserves pullbacks along $\mathcal{M}'$-morphisms because pullbacks along injective morphisms are constructed componentwise in $\mathbf{PTNets}$, hence Thm. 5.1 states that $(\mathbf{PTINets}, \mathcal{M}_{inj})$ is $\mathcal{M}$-adhesive. □

For PTI net transformation systems we require rules with rule morphisms in $\mathcal{M}$ and one of the following alternatives for matches:

1. general matches by PTI morphisms,
2. injective matches by $\mathcal{M}$,
3. token-injective matches by $\mathcal{M}' = \{f \in \mathbf{PTINets} | f_I \text{ inj.}\}$

For all three cases we have a suitable theory by DPO transformations with general, $\mathcal{M}$-matching resp. $\mathcal{M}'$-matching in $\mathcal{M}$-adhesive categories.

Theorem 5.3 ($\mathbf{AHLINets}(\mathbf{\Sigma})$ is $\mathcal{M}$-adhesive)**.**
For a fixed signature Σ *the category* $(\mathbf{AHLINets}(\mathbf{\Sigma}), \mathcal{M}_I)$ *is an* $\mathcal{M}$*-adhesive category where*

- $\mathbf{AHLINets}(\mathbf{\Sigma})$ *is the full subcategory of* **AHLINets** *containing all AHLI nets with the signature* Σ,
- $\mathcal{M}_I = \{f \in Mor_{\mathbf{AHLINets}(\mathbf{\Sigma})} | f_\Sigma = id_\Sigma, f_A \textit{ isomorphic, and } f_P, f_T, f_I \textit{ injective}\}$.

Proof. As shown in [Pra08], we already know that the category $(\mathbf{AHLNets}(SP), \mathcal{M}')$ of AHL nets over a specification SP is $\mathcal{M}$-adhesive with

$$\mathcal{M}' = \{f \in Mor_{\mathbf{AHLNets}(SP)} | f_A \text{ isomorphic, and } f_P, f_T \text{ injective}\}.$$

Given the functor $M : \mathbf{AHLNets}(\mathbf{\Sigma}, \emptyset) \to \mathbf{Sets}$ with

$$M(P, T, pre, post, cond, type, A) = A \otimes P,$$

we have $\mathbf{AHLINets}(\mathbf{\Sigma}) \sim \mathbf{ISystems}(\mathbf{AHLNets}(\mathbf{\Sigma}, \emptyset), M)$. M preserves pullbacks along $\mathcal{M}'$-morphisms because pullbacks along injective morphisms are constructed componentwise in $\mathbf{AHLNets}(\mathbf{\Sigma}, \emptyset)$ and we have only algebra isomorphisms in $\mathcal{M}'$. Hence, Thm. 5.1 states that $(\mathbf{AHLINets}(\mathbf{\Sigma}), \mathcal{M}_I)$ is $\mathcal{M}$-adhesive. □

Theorem 5.4 ($\mathbf{AHLINets}$ is $\mathcal{M}$-adhesive).
The category $(\mathbf{AHLINets}, \mathcal{M}_I)$ *is a* $\mathcal{M}$*-adhesive category where*

$$\mathcal{M}_I = \{f \in Mor_{\mathbf{AHLINets}} | f_\Sigma \textit{ injective }, f_A \textit{ isomorphic, and } f_P, f_T, f_I \textit{ injective}\}$$

Proof. As shown in [Pra08], we already know that the category of generalized AHL nets $(\mathbf{AHLNets}, \mathcal{M}')$ is $\mathcal{M}$-adhesive with

$$\mathcal{M}' = \{f \in Mor_{\mathbf{AHLNets}} | f_{SP} \text{ strict injective}, f_A \text{ isomorphic, and } f_P, f_T \text{ injective}\}.$$

We consider its full subcategory $(\mathbf{AHLNets}_\emptyset, \mathcal{M}'_{|\mathbf{AHLNets}_\emptyset})$ of generalized AHL nets with empty sets of specification equations, which by Thm. 2.3(i) from [Pra08] is itself $\mathcal{M}$-adhesive. Note that all injective morphisms between specifications without equations are strict.

Given the functor $M : \mathbf{AHLNets}_\emptyset \to \mathbf{Sets}$ with

$$M(SP, P, T, pre, post, cond, type, A) = A \otimes P,$$

we have $\mathbf{AHLINets} \sim \mathbf{ISystems}(\mathbf{AHLNets}_\emptyset, M)$. M preserves pullbacks along $\mathcal{M}'_{|\mathbf{AHLNets}_\emptyset}$-morphisms because pullbacks along injective morphisms are constructed componentwise in $\mathbf{AHLNets}_\emptyset$ and we have only algebra isomorphisms in $\mathcal{M}'$. Hence, Thm. 5.1 states that $(\mathbf{AHLINets}, \mathcal{M}_I)$ is $\mathcal{M}$-adhesive. □

6. Conclusion

In this article, we have introduced low- and high-level Petri nets with markings of individual tokens. The Petri net approach we presented is related to [vGP95, BMMS99] but in contrast the individual tokens in our framework are part of the syntactical definition of the Petri nets.

With the individuals being part of a net's syntax, we can consider Petri nets with an individual marking as objects of a category. Based on the double pushout transformation approach, we are able to define the rule based transformation for low- and high-level Petri nets with individual tokens. Important results are a necessary and sufficient condition that we give for the applicability of rules and other results that follow from the properties of $\mathcal{M}$-adhesive categories, because we are able to show that our Petri net categories are $\mathcal{M}$-adhesive, which is a short notion for weak adhesive HLR categories in [EEPT06].

The main advantage of the presented Petri net transformation approach over previous existing double pushout approaches (with "collective" markings) is that the marking of a net with individual tokens can be manipulated with rules, which is not possible in an adequate way with the "collective token approach". So, firing steps of Petri nets with individual tokens can be simulated with rules. Moreover, we proved a correspondence between such transition productions and firing steps.

We related the newly presented and existing net classes with functors and showed that they preserve enabling and firing.

7. Future Work

Although we have given some $\mathcal{M}$-adhesive categories of Petri nets with individual tokens, which allows us to use many interesting results for analysis of transformations, there are several useful additional properties and recent developments that are worthwhile to consider to examine if the presented categories comply to them.

Application Conditions In [EHL10], the results of [EEPT06] concerning parallel and concurrent rules have been lifted to transformation systems with rules with nested application conditions (see also [HP09]). The only additional property for $\mathcal{M}$-adhesive categories that is needed for these results is an $\mathcal{E}$-$\mathcal{M}$ factorization (and binary coproducts which we already have by cocompleteness).

Morphism pair factorization There are several useful theorems as the Concurrency Theorem and the Local Confluence Theorem, for which we need an $\mathcal{E}'$-$\mathcal{M}'$ pair factorization. Unfortunately, for the $\mathcal{M}$-adhesive categories in this article we can not simply apply the construction Thm. 4 from [PEL08] because for the individual marking functor M (cf. Def. 5.1 on page 41) establishing the constructed comma category does not in general yield an isomorphism $M(f)$ for a Petri net morphism f.

So for all these results, for transformation systems with or without application conditions, it may be convenient to directly prove an epi-$\mathcal{M}$ factorization, which (with binary coproducts) would directly constitute a $\mathcal{E}'$-$\mathcal{M}'$ pair factorization with $\mathcal{E}'$ as the class of jointly epimorphic morphisms.

Another possibility is to consider only nets with a finite structure, i.e. the nets with finite sets of places, transitions, and individuals, which is the kind of nets mostly used in practical cases. In an $\mathcal{M}$-adhesive category as given above, finite objects are the objects with a finite number of $\mathcal{M}$-subobjects, which in the case of nets are exactly the nets with finite structure. As shown in [BEGG10], the restriction of an $\mathcal{M}$-adhesive category to all its finite objects allows already to give a general construction for $\mathcal{E}$-$\mathcal{M}$ factorizations and for initial pushouts.

Rule amalgamation Another powerful concept is the amalgamation of rules (with application conditions) over a bundle of matches, which has been elaborated in [Gol10]. In order to instantiate the results in this article to our Petri net categories we need to show that they have so-called effective pushouts.

References

[AB09] Andrea Asperti and Nadia Busi. Mobile petri nets. *Mathematical Structures in Computer Science*, 19(6):1265–1278, 2009.

[BDHM06] P. Bottoni, F. De Rosa, K. Hoffmann, and M. Mecella. Applying Algebraic Approaches for Modeling Workflows and their Transformations in Mobile Networks. *Journal of Mobile Information Systems*, 2(1):51–76, 2006.

[BEE+09] Enrico Biermann, Hartmut Ehrig, Claudia Ermel, Kathrin Hoffmann, and Tony Modica. Modeling Multicasting in Dynamic Communication-based Systems by Reconfigurable High-level Petri Nets. In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2009, Corvallis, OR, USA, 20-24 September 2009, Proceedings*, pages 47–50. IEEE, 2009.

[BEGG10] Benjamin Braatz, Hartmut Ehrig, Karsten Gabriel, and Ulrike Golas. Finitary $\mathcal{M}$ -Adhesive Categories. In H. Ehrig, R. Heckel, G. Rozenberg, and G. Taentzer, editors, *Proc. International Conference on Graph Transformation (ICGT'10)*, volume 5214 of *LNCS*, Heidelberg, 2010. Springer. to appear.

[BMMS99] R. Bruni, J. Meseguer, U. Montanari, and V. Sassone. Functorial semantics for petri nets under the individual token philosophy. In *Category Theory and Computer Science, CTCS '99.*, volume 29 of *ENTCS*, page 18 pp. Elsevier, 1999.

[EBO92] H. Ehrig, M. Baldamus, and F. Orejas. New concepts for amalgamation and extension in the framework of specification logics. In *Proc. WADT-COMPASS-Workshop Dourdan, 1991*, pages 199–221. Springer LNCS 655, 1992.

[EEPT06] H. Ehrig, K. Ehrig, U. Prange, and G. Taentzer. *Fundamentals of Algebraic Graph Transformation.* EATCS Monographs in Theoretical Computer Science. Springer Verlag, 2006.

[EHL10] Hartmut Ehrig, Anngegret Habel, and Leen Lambers. Parallelism and Concurrency Theorems for Rules with Nested Application Conditions. *Electr. Communications of the EASST*, 26, 2010.

[EHP+08] H. Ehrig, K. Hoffmann, J. Padberg, C. Ermel, U. Prange, E. Biermann, and T. Modica. Petri Net Transformations. In *Petri Net Theory and Applications*, pages 1–16. I-Tech Education and Publication, 2008.

[EM85] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specification 1: Equations and Initial Semantics*, volume 6 of *EATCS Monographs on Theoretical Computer Science*. Springer, Berlin, 1985.

[EOP06] H. Ehrig, F. Orejas, and U. Prange. Categorical Foundations of Distributed Graph Transformation. In A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro, and G. Rozenberg, editors, *Proc. Third International Conference on Graph Transformation (ICGT'06)*, volume 4178 of *LNCS*, pages 215 – 229, Natal, Brazil, September 2006. Springer.

[EP97] C. Ermel and J. Padberg. Formalization of Variables in Algebraic High-Level Nets. Technical Report 97-19, Technical University Berlin, 1997.

[EPR93] H. Ehrig, J. Padberg, and L. Ribeiro. Algebraic high-level nets: Petri nets revisited. In *Proc. of the ADT-COMPASS Workshop'92 (Caldes de Malavella, Spain)*, Berlin, 1993. Technical Report TUB93-06.

[Gol10] U. Golas. Multi-Amalgamation in $\mathcal{M}$-Adhesive Categories. Technical Report 2010/05, Technische Universität Berlin, 2010.

[GR83] Ursula Goltz and Wolfgang Reisig. The Non-sequential Behavior of Petri Nets. *Information and Control*, 57(2/3):125–147, 1983.

[HB08] Awatef Hicheur and Kamel Barkaoui. Modelling collaborative workflows using recursive ecatnets. In *NOTERE '08: Proceedings of the 8th international conference on New technologies in distributed systems*, pages 1–11, New York, NY, USA, 2008. ACM.

[HME05] K. Hoffmann, T. Mossakowski, and H. Ehrig. High-Level Nets with Nets and Rules as Tokens. In *Proc. of 26th Intern. Conf. on Application and Theory of Petri Nets and other Models of Concurrency*, volume 3536 of *LNCS*, pages 268–288. Springer, 2005.

[HP00] Serge Haddad and Denis Poitrenaud. Modelling and analyzing systems with recursive petri nets. In *In Proc. of the Workshop on Discrete Event Systems - Analysis and Control*, pages 449–458. Kluwer Academics Publishers, 2000.

[HP09] Annegret Habel and Karl-Heinz Pennemann. Correctness of high-level transformation systems relative to nested conditions. *Mathematical Structures in Computer Science*, 19:1–52, 2009.

[KR07] Michael Köhler and Heiko Rölke. Dynamic transition refinement. *Electron. Notes Theor. Comput. Sci.*, 175(2):119–134, 2007.

[MEE$^+$10] Tony Modica, Claudia Ermel, Hartmut Ehrig, Kathrin Hoffmann, and Enrico Biermann. Modeling communication spaces with higher-order petri nets. In George E. Lasker and Jochen Pfalzgraf, editors, *Advances in Multiagent Systems, Robotics and Cybernetics: Theory and Practice*, volume

III, Tecumseh, Canada, 2010. The International Institute for Advanced Studies in Systems Research and Cybernetics. to appear.

[MM90] J. Meseguer and U. Montanari. Petri Nets are Monoids. *Information and Computation*, 88(2):105–155, 1990.

[Mod10] Tony Modica. Towards formal algebraic modeling and analysis of communication spaces. In Magne Haveraaen, Marina Lenisa, John Power, and Monika Seisenberger, editors, *CALCO Young Researchers Workshop (CALCO-jnr 2009) – Selected Papers*, number 5-2010 in Technical Report, pages 89–103. Università di Udine – Dipartimento di Matematica e Informatica, 2010.

[PEL08] U. Prange, H. Ehrig, and L. Lambers. Construction and Properties of Adhesive and Weak Adhesive High-Level Replacement Categories. *Applied Categorical Structures*, 16(3):365–388, 2008.

[PER95] J. Padberg, H. Ehrig, and L. Ribeiro. Algebraic high-level net transformation systems. *Mathematical Structures in Computer Science*, 5:217–256, 1995.

[Pra08] U. Prange. Towards algebraic high-level systems as weak adhesive HLR categories. In H. Ehrig, J. Pfalzgraf, and U. Prange, editors, *Proceedings of the ACCAT workshop at ETAPS 2007*, volume 203 / 6 of *Electronic Notes in Theoretical Computer Science*, pages 67–88, Amsterdam, 2008. Elsevier.

[Rei85a] Wolfgang Reisig. *Petri Nets: An Introduction*, volume 4 of *EATCS Monographs on Theoretical Computer Science*. Springer, 1985.

[Rei85b] Wolfgang Reisig. Petri Nets with Individual Tokens. *Theoretical Computer Science*, 41:185–213, 1985.

[Rei91] Wolfgang Reisig. Petri nets and algebraic specifications. *Theoretical Computer Science*, 80(1):1–34, March 1991.

[Syl09] Peggy Sylopp. Konzeption und Implementierung von Analysetechniken für rekonfigurierbare Objektnetze. Diplomarbeit, Technische Universität Berlin, 2009.

[TBG91] Andrzej Tarlecki, Rod M. Burstall, and Joseph A. Goguen. Some fundamental algebraic tools for the semantics of computation: Part 3: Indexed categories. *Theoretical Computer Science*, 91(2):239–264, 1991.

[Val78] Rüdiger Valk. Self-modifying nets, a natural extension of petri nets. In Giorgio Ausiello and Corrado Böhm, editors, *ICALP, Automata, Languages and Programming, Fifth Colloquium, Udine, Italy*, volume 62 of *Lecture Notes in Computer Science*, pages 464–476. Springer, 1978.

[vdABV+99] W. M. P. van der Aalst, T. Basten, H. M. W. Verbeek, P. A. C. Verkoulen, and M. Voorhoeve. Adaptive Workflow-On the Interplay between Flexibility and Support. In *Proc. of the Int. Conference on Enterprise Information Systems (ICEIS'99)*, pages 353–360, 1999.

[vG05] Robert Jan van Glabbeek. The individual and collective token interpretations of petri nets. In *Proceedings CONCUR 2005, 16 th International Conference on Concurrency Theory*, pages 323–337, London, UK, 2005. Springer-Verlag.

[vGP95] R. J. van Glabbeek and G. D. Plotkin. Configuration structures. In *LICS '95: Proceedings of the 10th Annual IEEE Symposium on Logic in Computer Science*, page 199, Washington, DC, USA, 1995. IEEE Computer Society.

[vGP09] R. J. van Glabbeek and G. D. Plotkin. Configuration structures, event structures and petri nets. *Theoretical Computer Science*, 410(41):4111–4159, 2009.

A. Appendix

A.1. Categorical Gluing Condition with Initial Pushouts

The following definitions, that can also be found in [EEPT06], provide notions to formulate an abstract categorical condition, the so-called gluing condition, which is necessary and sufficient for the (unique) existence of pushout complements in $\mathcal{M}$-adhesive categories. The gluing condition is used to show that the corresponding set-theoretical gluing conditions in the categories **PTNets** (see Def. 2.8 on page 10) and **AHLINets** (see Def. 3.8 on page 24) are necessary and sufficient conditions for the application of transformation rules $\varrho = (L \xleftarrow{l} K \xrightarrow{r} R)$ in suitable $\mathcal{M}$-adhesive categories $(\mathbf{PTINets}, \mathcal{M})$ (see Fact 2.11 on page 12) and $(\mathbf{AHLINets}, \mathcal{M})$ (see Fact 3.11 on page 27), respectively.

Definition A.1 (Boundary, Initial Pushout).
Given a morphism $f : L \to G$ in an $\mathcal{M}$-adhesive category, a morphism $b : B \to L$ with $b \in \mathcal{M}$ is called the *boundary* over f if there is a pushout complement of f and b such that (1) is a pushout which is *initial* over f. Initiality of (1) over f means, that for every pushout (2) with $b' \in \mathcal{M}$ there exist unique morphisms $b^* : B \to D$ and $c^* : C \to E$ with $b^*, c^* \in \mathcal{M}$ such that $b' \circ b^* = b, c' \circ c^* = c$ and (3) is a pushout. B is then called the *boundary object* and c the *context* with respect to f.

$$
\begin{tikzcd}
B \arrow[r, "b"] \arrow[d] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] \\
C \arrow[r, "c"'] & G
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, "b^*"] \arrow[d] \arrow[rr, bend left, "b"] \arrow[dr, phantom, "(\mathbf{3})"] & D \arrow[r, "b'"] \arrow[d] \arrow[dr, phantom, "(\mathbf{2})"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "c'"] & G
\end{tikzcd}
$$

Definition A.2 (Categorical Gluing Condition).
Let $l : K \to L \in \mathcal{M}$ and $f : L \to G$ be morphisms in a given $\mathcal{M}$-adhesive category $\mathbf{C}$ with initial pushouts.
We say that l and f satisfy the categorical gluing condition if for the initial pushout (1) over f there exists a morphism $b^* : B \to K$ such that $l \circ b^* = b$.

$$
\begin{tikzcd}
B \arrow[r, "b"'] \arrow[d, "g"'] \arrow[rr, bend left, "b^*"] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] & K \arrow[l, "l"] \arrow[r, "r"'] & R \\
C \arrow[r, "c"'] & G & &
\end{tikzcd}
$$

Given a production $\varrho = (L \xleftarrow{l} K \xrightarrow{r} R)$, we say ϱ and f satisfy the categorical gluing condition, iff l and f satisfy the categorical gluing condition.

Fact A.3 (Categorical Gluing Condition).
Given an adhesive HLR category $\mathbf{C}$ with initial pushouts, a match $f : L \to G$ satisfies the categorical gluing condition with respect to $l : K \to L \in \mathcal{M}$ (or a production

$\varrho = (L \xleftarrow{l} K \xrightarrow{r} R)$, respectively) if and only if the context object D exists, i.e. there is a pushout complement (2) of l and m:

$$
\begin{tikzcd}
B \arrow[r, "b"'] \arrow[rr, bend left, "b^*"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] \arrow[dr, phantom, "(\mathbf{2})"] & K \arrow[l, "l"'] \arrow[r, "r"'] \arrow[d, "k"] & R \\
C \arrow[r, "c"] \arrow[rr, bend right, "c^*"'] & G & D \arrow[l, "d"'] &
\end{tikzcd}
$$

If it exists, the context object D is unique up to isomorphism.

Proof. See Theorem 6.4 in [EEPT06]. □

In order to show initiality of a pushout (1) over a morphism $f : L \to G$ with corresponding opposite morphism $B \to C$ in an $\mathcal{M}$-adhesive category $\mathbf{C}$ (cf. Def. A.1 on the preceding page), one has to show for every pushout (2) over f with corresponding opposite morphism $D \to E$, that there exist unique morphisms $b^* : B \to D$ and $c^* : C \to E$ forming a pushout (cf. Def. A.1). The following Lemma states the fact, that the existence of the required morphism c^* induces the remaining requirements.

Lemma A.1 (Morphism-Pushout-Lemma).
Given pushouts (1) and (2) in an $\mathcal{M}$-adhesive category $\mathbf{C}$ where $b, d \in \mathcal{M}$.
If there is a morphism $c^* : C \to E$ with $c = e \circ c^*$ then this c^* is unique and $c^* \in \mathcal{M}$ and there exists a unique $b^* : B \to D \in \mathcal{M}$ with $b = d \circ b^*$ such that (3) is pushout in $\mathbf{C}$.

$$
\begin{tikzcd}
B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"'] \arrow[dr, phantom, "(\mathbf{2})"] & D \arrow[l, "d"'] \arrow[d, "h"] \\
C \arrow[r, "c"] \arrow[rr, bend right, "c^*"'] & G & E \arrow[l, "e"']
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, "b^*"'] \arrow[rr, bend left, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{3})"] & D \arrow[r, "d"'] \arrow[d, "h"'] \arrow[dr, phantom, "(\mathbf{2})"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "e"] & G
\end{tikzcd}
$$

Proof. We use in the following that also $c, e \in \mathcal{M}$ due to pushouts (1) and (2) and the fact that $\mathcal{M}$-morphisms are closed under pushouts.

uniqueness of c^*:
Let $\bar{c} : C \to E$ with $c = e \circ \bar{c}$. Then there is

$$e \circ \bar{c} = c = e \circ c^*$$

which by the fact that e is a monomorphism implies that $\bar{c} = c^*$.

unique morphism b^*:
Since $d \in \mathcal{M}$ the pushout (2) is also a pullback and since (1) is a pushout there is

$$f \circ b = c \circ g = e \circ c^* \circ g$$

from the pullback property follows that there is a unique morphism $b^* : B \to D$ with

$$c^* \circ g = h \circ b^* \text{ and } b = d \circ b^*$$

leading to commuting diagram (3).

$$\begin{tikzcd}
B \arrow[r, "b^*"'] \arrow[d, "g"'] \arrow[rr, bend left, "b"] \arrow[dr, phantom, "\mathbf{(3)}"] & D \arrow[r, "d"'] \arrow[d, "h"'] \arrow[dr, phantom, "\mathbf{(2)}"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "e"] & G
\end{tikzcd}$$

$b^*, c^* \in \mathcal{M}$:

The morphisms b and c are $\mathcal{M}$-morphisms. So the fact that

$$b = d \circ b^* \text{ and } c = e \circ c^*$$

together with the fact that $\mathcal{M}$-morphisms are closed under decomposition implies that $b^*, c^* \in \mathcal{M}$.

(3) is pushout:

Since c is an $\mathcal{M}$-morphism and $\mathcal{M}$-morphisms are closed under decomposition due to the fact that $c \circ id_C = c \in \mathcal{M}$ there is $id_C \in \mathcal{M}$. Consider the following cube:

$$\begin{tikzcd}
 & C \arrow[rr, "c^*"] \arrow[dd, "id_C", near start] & & E \arrow[rr, "id_E"] \arrow[dd, "id_E", near start] & & E \arrow[dd, "e"] \\
B \arrow[ur, "g"] \arrow[rr, "b^*", near end, crossing over] \arrow[dd, "id_B"'] & & D \arrow[ur, "h"] \arrow[rr, "id_D", near start, crossing over] & & D \arrow[ur, "h"] & \\
 & C \arrow[rr, "c^*", near end] & & E \arrow[rr, "e", near end] & & G \\
B \arrow[ur, "g"] \arrow[rr, "b^*"] & & D \arrow[ur, "h"] \arrow[rr, "d"] \arrow[from=uu, "id_D", near start, crossing over] & & L \arrow[ur, "f"] \arrow[from=uu, "d", near start, crossing over] &
\end{tikzcd}$$

- The left face is a pullback,
- the right face is pullback (2),
- the front left and back left faces are pullbacks,
- the front right and back right faces are pullbacks because $d, e \in \mathcal{M}$,
- the bottom is the pushout (1),
- and the morphisms id_C, d and e are $\mathcal{M}$-morphisms.

So the cube is a weak Van Kampen cube implying that the top face (3) of the cube is a pushout in $\mathbf{C}$.

□

In the following two lemmas we show, that the constructions $(\_^\sharp \otimes \_)^\oplus$ and $\mathcal{P}_{fin}(\_^\sharp)$, which are used in the pre and post conditions, and conditions of AHLI nets, respectively, are compositional. We need these properties for the proofs of Fact 3.10 on page 27 and Fact 3.11 on page 27.

Lemma A.2 (Compositionality of $(\_^\sharp \otimes \_)^\oplus$)**.**
Given three AHLI nets

$$AN_i = (\Sigma_i, P_i, T_i, pre_i, post_i, cond_i, type_i, I_i, A_i) \text{ where } (i \in \{1, 2, 3\})$$

together with signature morphisms $f_\Sigma : \Sigma_1 \to \Sigma_2$*,* $g_\Sigma : \Sigma_2 \to \Sigma_3$ *and functions* $f_P : P_1 \to P_2$*,* $g_P : P_2 \to P_3$ *with*

$$type_2 \circ f_P = f_S \circ type_1$$

Then for all

$$\sum_{i=1}^{n} (term_i, p_i) \in (T_{OP_1}(X_1) \otimes P_1)^\oplus$$

there is

$$(g_\Sigma^\sharp \otimes g_P)^\oplus((f_\Sigma^\sharp \otimes f_P)^\oplus(\sum_{i=1}^{n} (term_i, p_i))) = ((g_\Sigma \circ f_\Sigma)^\sharp \otimes (g_P \circ f_P))^\oplus(\sum_{i=1}^{n} (term_i, p_i))$$

Proof. Let

$$\sum_{i=1}^{n} (term_i, p_i) \in (T_{OP_1}(X_1) \otimes P_1)^\oplus$$

Due to the freeness of $T_{OP_1}(X_1)$ there is

$$V_{f_\Sigma}(g_\Sigma^\sharp) \circ f_\Sigma^\sharp = (f_\Sigma \circ g_\Sigma)^\sharp$$

So we have

$$
\begin{aligned}
& (g_\Sigma^\sharp \otimes g_P)^\oplus((f_\Sigma^\sharp \otimes f_P)^\oplus(\sum_{i=1}^{n}(term_i, p_i))) \\
= \; & (g_\Sigma^\sharp \otimes g_P)^\oplus(\sum_{i=1}^{n}((f_\Sigma^\sharp)_{type_1(p_i)}(term_i), f_P(p_i))) \\
= \; & \sum_{i=1}^{n}((g_\Sigma^\sharp)_{type_2(f_P(p_i))}((f_\Sigma^\sharp)_{type_1(p_i)}(term_i)), g_P(f_P(p_i))) \\
= \; & \sum_{i=1}^{n}((g_\Sigma^\sharp)_{f_S(type_1(p_i))}((f_\Sigma^\sharp)_{type_1(p_i)}(term_i)), g_P(f_P(p_i))) \\
= \; & \sum_{i=1}^{n}(V_{f_\Sigma}(g_\Sigma^\sharp)_{type_1(p_i)}((f_\Sigma^\sharp)_{type_1(p_i)}(term_i)), g_P(f_P(p_i))) \\
= \; & \sum_{i=1}^{n}((V_{f_\Sigma}(g_\Sigma^\sharp) \circ f_\Sigma^\sharp)_{type_1(p_i)}(term_i), g_P(f_P(p_i))) \\
= \; & \sum_{i=1}^{n}((g_\Sigma \circ f_\Sigma)^\sharp_{type_1(p_i)}(term_i), g_P \circ f_P(p_i)) \\
= \; & ((g_\Sigma \circ f_\Sigma)^\sharp \otimes (g_P \circ f_P))^\oplus(\sum_{i=1}^{n}(term_i, p_i))
\end{aligned}
$$

□

Lemma A.3 (Compositionality of $\mathcal{P}_{fin}(\_^\sharp)$).
Given three AHLI nets

$$AN_i = (\Sigma_i, P_i, T_i, pre_i, post_i, cond_i, type_i, I_i, A_i) \text{ where } (i \in \{1, 2, 3\})$$

together with signature morphisms $f_\Sigma : \Sigma_1 \to \Sigma_2$ *and* $g_\Sigma : \Sigma_2 \to \Sigma_3$.
Then for all $E \in \mathcal{P}_{fin}(Eqns(\Sigma_1))$ *there is*

$$\mathcal{P}_{fin}(g_\Sigma^\sharp)(\mathcal{P}_{fin}(f_\Sigma^\sharp)(E)) = \mathcal{P}_{fin}((g_\Sigma \circ f_\Sigma)^\sharp)(E)$$

Proof. Let $E \in \mathcal{P}_{fin}(Eqns(\Sigma_1))$. For $e = (t_l, t_r) \in Eqns(\Sigma_1)$ the extension $f_\Sigma^\sharp$ of a signature morphism $f_\Sigma : \Sigma_1 \to \Sigma_2$ to equations of Σ_1 is defined by

$$f_\Sigma^\sharp(e) = ((f_\Sigma^\sharp)_s(t_l), (f_\Sigma^\sharp)_s(t_r))$$

where s is the sort of terms t_l and t_r, i.e. $t_l, t_r \in T_{OP_1}(X_1)_s$. The function $(f_\Sigma^\sharp)_s$ on the right hand side of the equation is the extension of f_Σ to terms of type s, i.e.

$$(f_\Sigma^\sharp)_s : T_{OP_1}(X_1)_s \to V_{f_\Sigma}(T_{OP_2}(X_2))_s$$

Due to the definition of the forgetful functor V_{f_Σ} there is

$$V_{f_\Sigma}(T_{OP_2}(X_2))_s = T_{OP_2}(X_2)_{f_S(s)}$$

and hence there is

$$(f_\Sigma^\sharp)_s : T_{OP_1}(X_1)_s \to T_{OP_2}(X_2)_{f_S(s)}$$

So we have

$$\begin{aligned}
\mathcal{P}_{fin}(g_\Sigma^\sharp)(\mathcal{P}_{fin}(f_\Sigma^\sharp)(E)) &= \mathcal{P}_{fin}(g_\Sigma^\sharp)(\mathcal{P}_{fin}(f_\Sigma^\sharp)(\{e \mid e \in E\})) \\
&= \mathcal{P}_{fin}(g_\Sigma^\sharp)(\{((f_\Sigma^\sharp)_s(t_l), (f_\Sigma^\sharp)_s(t_l)) \mid (t_l, t_r) \in E\}) \\
&= \{((g_\Sigma^\sharp)_{f_S(s)}((f_\Sigma^\sharp)_s(t_l)), (g_\Sigma^\sharp)_{f_S(s)}((f_\Sigma^\sharp)_s(t_r))) \mid (t_l, t_r) \in E\} \\
&= \{(V_{f_\Sigma}(g_\Sigma^\sharp)_s((f_\Sigma^\sharp)_s(t_l)), V_{f_\Sigma}(g_\Sigma^\sharp)_s((f_\Sigma^\sharp)_s(t_r))) \mid (t_l, t_r) \in E\} \\
&= \{((V_{f_\Sigma}(g_\Sigma^\sharp) \circ f_\Sigma^\sharp)_s(t_l), (V_{f_\Sigma}(g_\Sigma^\sharp) \circ f_\Sigma^\sharp)_s(t_r)) \mid (t_l, t_r) \in E\} \\
&= \{((g_\Sigma \circ f_\Sigma)_s^\sharp(t_l), (g_\Sigma \circ f_\Sigma)_s^\sharp(t_r)) \mid (t_l, t_r) \in E\} \\
&= \mathcal{P}_{fin}((g_\Sigma \circ f_\Sigma)^\sharp)(E)
\end{aligned}$$

□

In the following two lemmas we show, that the constructions $\_ \otimes \_$ and $\_^\sharp$ preserve monomorphisms, which is used in the proofs of Fact 3.10 on page 27 and Fact 3.11 on page 27.

Lemma A.4 ($\_ \otimes \_$ preserves Monomorphisms)**.**
Given two AHLI nets

$$AN_i = (\Sigma_i, P_i, T_i, pre_i, post_i, cond_i, type_i, I_i, A_i) \text{ where } (i \in \{1, 2\})$$

and a monomorphism $f = (f_P, f_T, f_\Sigma, f_A, f_I) : AN_1 \to AN_2$ in **AHLINets**.
Then $f_A \otimes f_P : A_1 \otimes P_1 \to A_2 \otimes P_2$ is a monomorphism in **Sets**.

Proof. Let $(a_1, p_1), (a_2, p_2) \in A_1 \otimes P_1$ with

$$(f_A \otimes f_P)(a_1, p_1) = (a, p) = (f_A \otimes f_P)(a_2, p_2)$$

Then there is

$$\begin{aligned}
(f_{A,type_1(p_1)}(a_1), f_P(p_1)) &= (f_A \otimes f_P)(a_1, p_1) \\
&= (a, p) \\
&= (f_A \otimes f_P)(a_2, p_2) \\
&= (f_{A,type_1(p_2)}(a_2), f_P(p_1))
\end{aligned}$$

which means that

$$f_{A,type_1(p_1)}(a_1) = a = f_{A,type_1(p_2)}(a_2)$$

and

$$f_P(p_1) = p = f_P(p_2)$$

Since f is a monomorphism in **AHLINets** the function f_P is injective implying that $p_1 = p_2$. This means that

$$f_{A,type_1(p_2)}(a_1) = f_{A,type_1(p_1)}(a_1) = a = f_{A,type_1(p_2)}(a_2)$$

and since f is a monomorphism in **AHLINets** there is $f_{A,type_1(p_2)}$ injective implying that $a_1 = a_2$. Hence there is $(a_1, p_1) = (a_2, p_2)$ implying that $f_A \otimes f_P$ is injective, i.e. it is a monomorphism in **Sets**. □

Lemma A.5 ($\_^\sharp$ preserves Monomorphisms).
Given a signature morphism $f : \Sigma_1 \to \Sigma_2$.
If f is a monomorphism in **Sig** *then $f^\sharp : T_{OP_1}(X_1) \to V_f(T_{OP_2}(X_2))$ is a monomorphism in* $\mathbf{Alg(\Sigma_1)}$.

Proof. Since monomorphisms in $\mathbf{Alg(\Sigma_1)}$ are exactly the injective homomorphisms it is sufficient to show that $f^\sharp$ is injective. Let $t_1, t_2 \in T_{OP_1}(X_1)_s$ with

$$f^\sharp_s(t_1) = t = f^\sharp_s(t_2)$$

The recursive definition of terms allows us to prove this property by a structural induction over $t \in V_f(T_{OP_2}(X_2))_s = T_{OP_2}(X_2)_{f_S(s)}$.

Basis.

- $t = x \in X_{2,f_S(s)}$
  This means that there are $x_1, x_2 \in X_{1,s}$ with

$$\begin{aligned} & f^\sharp_s(x_1) = x = f^\sharp_s(x_2) \\ \Leftrightarrow\quad & f_X(x_1) = x = f_X(x_2) \end{aligned}$$

  and since f_X is injective this means $t_1 = x_1 = x_2 = t_2$.
- $t = c$ with $c :\to f_S(s) \in OP_2$
  This means that for $i = 1, 2$ there are $c_i :\to s \in OP_1$ with

$$\begin{aligned} & f^\sharp_s(c_1) = c = f^\sharp_s(c_2) \\ \Leftrightarrow\quad & f_{OP}(c_1) = c = f_{OP}(c_2) \end{aligned}$$

  and since f_{OP} is injective this means $t_1 = c_1 = c_2 = t_2$.

Hypothesis. For $t^i \in V_f(T_{OP_2}(X_2))_{s_i} = T_{OP_2}(X_2)_{f_S(s_i)}$ $(i = 1, \ldots, n)$ there is

$$f^\sharp_{s_i}(t_1) = t^i = f^\sharp_{s_i}(t_2) \Rightarrow t_1 = t_2$$

Step. Let $t = op(t^1, \dots, t^n)$ with $op : f_S(s_1) \dots f_S(s_n) \to f_S(s) \in OP_2$ and $t^i \in T_{OP_2}(X_2)_{f_S(s_i)}$ $(i = 1, \dots, n)$.
Then from $f_s^\sharp(t_1) = t = f_s^\sharp(t_2)$ follows that there are $op_i : s_1 \dots s_n \to s \in OP_1$ $(i = 1, 2)$ such that

$$
\begin{aligned}
& f_s^\sharp(t_1) = t = f_s^\sharp(t_2) \\
\Leftrightarrow\ & f_s^\sharp(op_1(t_1^1, \dots, t_1^n)) = op(t^1, \dots, t^n) = f_s^\sharp(op_2(t_2^1, \dots, t_2^n)) \\
\Leftrightarrow\ & f_{OP}(op_1)(f_{s_1}^\sharp(t_1^1), \dots, f_{s_n}^\sharp(t_1^n)) = op(t^1, \dots, t^n) = f_{OP}(op_2)(f_{s_1}^\sharp(t_2^1), \dots, f_{s_n}^\sharp(t_2^n)) \\
\Leftrightarrow\ & f_{OP}(op_1) = op = f_{OP}(op_2) \wedge \forall i \in \{1, \dots, n\} : f_{s_i}^\sharp(t_1^i) = t^i = f_{s_i}^\sharp(t_2^i)
\end{aligned}
$$

Due to the injectivity of f_{OP} there is $op_1 = op_2$ and by the induction hypothesis there is $t_1^i = t_2^i$ for all $i \in \{1, \dots, n\}$. Hence we have $t_1 = t_2$.

□

A.2. Initial Pushouts in Sets

In this section we define a gluing condition in the category **Sets** and show, that the satisfaction of the condition for a suitable $\mathcal{M}$-adhesive category (**Sets**, $\mathcal{M}$) is equivalent to the categorical gluing condition. This provides a necessary and sufficient condition for the existence of (unique) pushout complements in **Sets**, which is used in the corresponding facts of the set-based categories **PTINets** and **AHLINets**.

Definition A.4 (Gluing Condition in **Sets**).
Let $l : K \to L$ and $f : L \to G$ be morphisms in **Sets** with $l \in \mathcal{M}$.
We define the set of identification points

$$IP = \{x \in L \mid \exists x' \neq x : f(x) = f(x')\}$$

and the set of gluing points

$$GP = l(K)$$

We say that l and f satisfy the gluing condition if $IP \subseteq GP$.

The following lemma provides a set-theoretical construction of pushout complements in **Sets**, whereas the category-theoretical construction of pushout complements is defined via a pushout over the boundary (see Theorem 6.4 in [EEPT06]).

Lemma A.6 (Pushout Complement in **Sets**).
Let $l : K \to L$ and $f : L \to G$ be morphisms in **Sets**.
There is a pushout complement C of l and f, if l and f satisfy the gluing condition.
If a pushout complement exists it can be computed by

$$C = (G \setminus f(L)) \cup f(l(K))$$

together with inclusion $c : C \to G$ and a morphism $g : K \to C$ with $g(x) = f(l(x))$ for every $x \in K$.

$$\begin{tikzcd}
L \arrow[d, "f"'] \arrow[dr, phantom, "(\mathbf{1})"] & K \arrow[l, "l"'] \arrow[d, "g"] \\
G & C \arrow[l, "c"]
\end{tikzcd}$$

Proof. We define C, c and g as above. We have to show that (1) is pushout in **Sets**.

commutativity of (1):
Let $x \in L$.

$$f(l(x)) = g(x) = c(g(x))$$

universal property:
Let H be a set together with morphisms $c' : C \to H$ and $f' : L \to H$ with

$$c' \circ g = f' \circ l$$

We define a morphism $h : G \to H$ with

$$h(x) = \begin{cases} c'(x) & \text{, if } x \in C; \\ f'(x') & \text{, for } f(x') = x \text{ otherwise.} \end{cases}$$

For the well-definedness of h we have to check if for every $x \in G$ with $x \notin C$ there is a unique $x' \in L$ such that $f(x') = x$.

$$\begin{aligned} x \notin C &\overset{\text{def. } C}{\Leftrightarrow} x \notin (G \setminus f(L)) \cup f(l(K)) \\ &\Leftrightarrow \neg x \in (G \setminus f(L)) \cup f(l(K)) \\ &\Leftrightarrow \neg(x \in G \setminus f(L) \vee x \in f(l(K))) \\ &\Leftrightarrow x \notin G \setminus f(L) \wedge x \notin f(l(K)) \end{aligned}$$

From the fact that $x \in G$ and $x \notin G \setminus f(L)$ we have $x \in f(L)$ which means that there is $x' \in L$ with $f(x') = x$.
Let us assume that x' is not unique, i.e. there is $x'' \in L$ with $x' \neq x''$ and $f(x'') = x$. Then x' is an identification point which implies that $x' \in GP = f(l(K))$ because l and f satisfy the gluing condition. This is a contradiction and hence x' is unique.

$$\begin{tikzcd}
 & L \arrow[d, "f"] \arrow[dr, phantom, "(\mathbf{1})"] \arrow[ddl, bend right, "f'"'] \arrow[dl, phantom, "(\mathbf{2})"] & K \arrow[l, "l"'] \arrow[d, "g"] \\
 & G \arrow[dl, "h"] \arrow[d, phantom, "(\mathbf{3})"] & C \arrow[l, "c"] \arrow[dll, bend left, "c'"] \\
H & &
\end{tikzcd}$$

Let $x \in C$. Then there is

$$h \circ c(x) = h(x) = c'(x)$$

which means that diagram (2) commutes.
Let $x \in L$. Then we distinguish the following cases:

- **Case 1**: $x \in l(K)$
  Then there is $k \in K$ with $l(k) = x$ and

$$h(f(x)) = h(f(l(k))) = h(c(g(k))) = c'(g(k)) = f'(l(k)) = f'(x)$$

- **Case 2**: $x \notin l(K)$
  Then there is $f(x) \notin C$ and therefore

$$h(f(x)) = f'(x)$$

Hence diagram (3) commutes.
For the uniqueness of h let $h' : G \to H$ with $h' \circ c = c'$ and $h' \circ f = f'$.
Let $x \in G$.

- **Case 1**: $x \in C$

$$h'(x) = h'(c(x)) = c'(x) = h(x)$$

- **Case 2**: $x \notin C$
  As we have shown above in this case there is a unique $x' \in L$ with $f(x') = x$ and we obtain

$$h'(x) = h'(f(x')) = f'(x') = h(x)$$

So we have for all $x \in G$ that $h'(x) = h(x)$ and hence $h' = h$.

□

In the following definition and facts we define the boundary and initial pushout over a morphism $f : L \to G$ in **Sets** and show that the satisfaction of the gluing condition in **Sets** is equivalent to the satisfaction of the categorical gluing condition in an $\mathcal{M}$-adhesive category $(\mathbf{Sets}, \mathcal{M})$ where the class of monomorphisms $\mathcal{M}$ contains inclusions. A suitable class $\mathcal{M}$ is the class containing all monomorphisms in **Sets**, because **Sets** is an adhesive category (see Theorem 4.6 in [EEPT06]).

Definition A.5 (Boundary in **Sets**).
Given a morphism $f : L \to G$ in **Sets**. The boundary B of f is the set $B = IP$ of identification points together with an inclusion $b : B \to L$.

Fact A.6 (Initial Pushout in **Sets**).
Given a morphism $f : L \to G$ in **Sets**, the boundary B of f and the pushout complement C of f and b, defined as

$$C = (G \setminus f(L)) \cup f(l(K))$$

together with inclusion $c : C \to G$ and a morphism $g : K \to C$ with $g(x) = f(l(x))$ for every $x \in K$.

Then diagram (1) is initial pushout in **Sets**.

$$\begin{tikzcd} B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(1)"] & L \arrow[d, "f"] \\ C \arrow[r, "c"'] & G \end{tikzcd}$$

Proof.

(1) is pushout:
The fact that (1) is pushout follows from Lemma A.6 by the fact that b and f satisfy the gluing condition. It remains to show that (1) is initial.
Let (2) be a pushout in **Sets** with $d \in \mathcal{M}$.

$$\begin{tikzcd} B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(1)"] & L \arrow[d, "f"] \arrow[dr, phantom, "(2)"] & D \arrow[l, "d"'] \arrow[d, "h"] \\ C \arrow[r, "c"'] & G & E \arrow[l, "e"] \end{tikzcd}$$

function c^*:
We define a function $c^* : C \to E$ with

$$c^*(x) = y \text{ with } e(y) = c(x)$$

well-definedness of c^*:
For the well-definedness of c^* we have to show that for every $x \in C$ there is a unique $y \in E$ with $e(y) = c(x)$.
We distinguish the following cases for $x \in C$:

- **Case 1**: $x \notin f(L)$
  There is $c(x) = x \in G$. Since (2) is pushout in **Sets** the functions f and e are jointly surjective which by the fact that there is no $y \in L$ with $f(y) = x$ implies that there is $y \in E$ with $e(y) = x = c(x)$.
- **Case 2**: $x \in f(b(B))$
  Then there is $z \in B$ with $f(b(z)) = x$. From $z \in B = IP$ and the fact that E is a pushout complement of d and f follows that $z \in d(D)$, i.e. there is $z' \in D$ with $d(z') = z$. Let $y = h(z')$. Then we have

$$e(y) = e(h(z')) = f(d(z')) = f(z) = f(b(z)) = x = c(x)$$

So for every $x \in C$ there is a suitable element $y \in E$ with $e(y) = c(x)$. Since $d \in \mathcal{M}$ and pushouts in **Sets** are closed under $\mathcal{M}$-morphisms there is also $e \in \mathcal{M}$, i.e. e is injective which implies the uniqueness of y.
So c^* is well-defined. The fact that $e \circ c^* = c$ follows directly from the definition of c^*.

uniqueness of c^*, existence of b^* and pushout:
By Lemma A.1 the morphism c^* is the unique morphism with $c = e \circ c^*$ and there is a unique morphism $b^* : B \to D$ with $b = d \circ b^*$ such that (3) is a pushout in **Sets**.

$$\begin{tikzcd}
B \arrow[r, "b^*"'] \arrow[rr, bend left, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "\mathbf{(3)}"] & D \arrow[r, "d"'] \arrow[d, "h"'] \arrow[dr, phantom, "\mathbf{(2)}"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "e"] & G
\end{tikzcd}$$

Hence diagram (1) is an initial pushout in **Sets**. □

Fact A.7 (Characterization of Gluing Condition in **Sets**).
Let $l : K \to L$ and $f : L \to G$ be morphisms in **Sets** with $l \in \mathcal{M}$.
The morphisms l and f satisfy the gluing condition in **Sets** if and only if they satisfy the categorical gluing condition.

Proof.

If. Let l and f satisfy the categorical gluing condition, i.e. for initial pushout (1) there is a function $b^* : B \to K$ with $l \circ b^* = b$.

$$\begin{tikzcd}
B \arrow[r, "b"'] \arrow[rr, bend left, "b^*"] \arrow[d, "g"'] \arrow[dr, phantom, "\mathbf{(1)}"] & L \arrow[d, "f"] & K \arrow[l, "l"] \\
C \arrow[r, "c"'] & G &
\end{tikzcd}$$

Let $x \in IP$. Then there is $x \in B$ and $y = b^*(x) \in K$ with

$$l(y) = l(b^*(x)) = b(x) = x$$

which means that $x \in l(K) = GP$. Hence l and f satisfy the gluing condition in **Sets**.

Only If. Let l and f satisfy the gluing condition in **Sets**, i.e. there is $IP \subseteq GP = l(K)$. We define a function $b^* : B \to K$ with

$$b^*(x) = l^{-1}(x)$$

For every $x \in B$ there is $x \in IP$ which by the fact that l and f satisfy the gluing condition implies that $x \in GP = l(K)$, i.e. there is a preimage of x with respect to l. The preimage is unique since l is injective because $l \in \mathcal{M}$. Hence b is well-defined and there is

$$l(b^*(x)) = l(l^{-1}(x)) = x = b(x)$$

which means that l and f satisfy the categorical gluing condition.

□

A.3. Initial Pushouts in PTNets

The construction and proof of the gluing condition for PTI nets (see Fact 2.12 on page 12) allows to derive the corresponding result for P/T nets as a special case where the set of individual tokens is empty.

Definition A.8 (Gluing Condition in **PTNets**).
Given P/T nets K, L and G and P/T morphisms $l : K \to L$ and $f : L \to G$. We define the set of identification points

$$IP = IP_P \cup IP_T$$

with

- $IP_P = \{x \in P_L \mid \exists x' \neq x : f_P(x) = f_P(x')\}$,
- $IP_T = \{x \in T_L \mid \exists x' \neq x : f_T(x) = f_T(x')\}$,

the set of dangling points

$$DP = \{p \in P_L \mid \exists t \in T_G \setminus f_T(T_L) : f_P(p) \in ENV(t)\}$$

and the set of gluing points

$$GP = l_P(P_K) \cup l_T(T_K)$$

We say that l and f satisfy the gluing condition if $IP \cup DP \subseteq GP$.

Definition A.9 (Boundary in **PTNets**).
Given a morphism $f : L \to G$ in **PTNets**. The boundary of f is a P/T net

$$B = (P_B, T_B, pre_B, post_B)$$

with

- $P_B = DP_T \cup IP_P \cup P_{IP_T}$
- $P_{IP_T} = \{p \in P_L \mid \exists t \in IP_T : p \in ENV(t)\}$
- $T_B = IP_T$

- $pre_B(t) = pre_L(t)$
- $post_B(t) = post_L(t)$

together with an inclusion $b : B \to L$.

Well-definedness.

$pre_B, post_B : T_B \to P_B^{\oplus}$:
Let $t \in T_B$ and let $p \leq pre_B(t)$. Then there is $p \leq pre_L(t)$ which means that $p \in P_L$. Then there is $t \in IP_T$ which, by the fact that $p \in ENV(t)$, means that $p \in P_{IP_T} \subseteq P_B$.
So pre_B is well-defined. The proof for $post_B$ works completely analogously.

inclusion $b : B \to L$:
We obtain an inclusion morphism $b : B \to L$ from the fact that pre_B and $post_B$ are restrictions of the respective functions in L.

□

Fact A.10 (Initial Pushout in **PTNets**).
Given a morphism $f : L \to G$ in **PTNets**, the boundary B of f and the P/T net $C = (P_C, T_C, pre_C, post_C)$ with

- $P_C = (P_G \setminus f_P(P_L)) \cup f_P(b_P(P_B))$
- $T_C = (T_G \setminus f_T(T_L)) \cup f_T(b_T(T_B))$
- $pre_C(t) = pre_G(t)$
- $post_C(t) = post_G(t)$

Then diagram (1) where $g := f|_B$ is initial pushout in **PTNets**.

$$\begin{tikzcd} B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(1)"] & L \arrow[d, "f"] \\ C \arrow[r, "c"'] & G \end{tikzcd}$$

Proof. Analogously to Fact 2.10 with $I = \emptyset$. □

Fact A.11 (Characterization of Gluing Condition in **PTNets**).
Let $l : K \to L$ and $f : L \to G$ be morphisms in **PTNets** with $l \in \mathcal{M}$.
The morphisms l and f satisfy the gluing condition in **PTNets** if and only if they satisfy the categorical gluing condition.

Proof. Analogously to Fact 2.11 with $I = \emptyset$. □

B. Proofs

B.1. Proof of Fact 2.10

In this section we prove the well-definedness of the PTI net C defined in Fact 2.10 and that the construction leads to an initial pushout in the category **PTINets**.

Proof.

well-definedness of C:

$pre_C, post_C : T_C \to P_C^{\oplus}$:
Follows from the well-definedness of C in Fact A.10 and the fact that we have the same set of transitions, and the set of places in Fact A.10 is a subset of P_C.

$m_C : I_C \to P_C$:
Let $i \in I_C$. Then there is $i \in (I_{NI} \setminus f_I(I_L)) \cup f_I(b_I(I_B))$.

Case 1: $i \notin f_I(I_L)$

Case 1.1: $\exists p \in P_L : f_P(p) = m_{NI}(i)$
This means that $p \in DP$ and therefore $p \in P_B$ and $f_P(p) \in P_C$. So we have

$$m_C(i) = m_{NI}(i) = f_P(p) \in P_C$$

Case 1.2: $\nexists p \in P_L : f_P(p) = m_{NI}(i)$
This means that $m_{NI}(i) \notin f_P(P_L)$ and hence $m_C(i) = m_{NI}(i) \in P_C$.

Case 2: $i \in f_I(b_I(I_B))$
Then there exists $j \in I_B$ with $f_I(b_I(j)) = i$ and we have for $m_C(i) = m_{NI}(i)$ that

$$m_{NI}(i) = m_{NI}(f_I(b_I(j))) = f_P(m_L(b_I(j))) = f_P(b_P(m_B(j)))$$

and since $f_P(b_P(P_B)) \subseteq P_C$ there is $m_C(i) \in P_C$.

well-definedness of c:
We obtain an inclusion morphism $c : C \to NI$ from the fact that $pre_C, post_C$ and m_C are restrictions of the respective functions in NI.

well-definedness of g:
For $J = \{P, T, I\}$ we obtain well-defined functions $g_J : J_B \to J_C$ because for $j \in J_B$ there is

$$g_J(j) = f_J(j) = f_J(b_J(j)) \in J_C$$

The morphism g preserves pre and post domains and markings because it is a restriction of f which is a well-defined PTI morphism.

(1) is pushout:

Due to Lemma A.6 the diagrams (2)-(4) are pushouts in **Sets** which implies that (1) is pushout in **PTINets** because the pushout in **PTINets** can be constructed componentwise.

$$\begin{tikzcd} P_B \arrow[r, "b_P"] \arrow[d, "g_P"'] \arrow[dr, phantom, "(\mathbf{2})"] & P_L \arrow[d, "f_P"] \\ P_C \arrow[r, "c_P"'] & P_{NI} \end{tikzcd} \quad \begin{tikzcd} T_B \arrow[r, "b_T"] \arrow[d, "g_T"'] \arrow[dr, phantom, "(\mathbf{3})"] & T_L \arrow[d, "f_T"] \\ T_C \arrow[r, "c_T"'] & T_{NI} \end{tikzcd} \quad \begin{tikzcd} I_B \arrow[r, "b_I"] \arrow[d, "g_I"'] \arrow[dr, phantom, "(\mathbf{4})"] & I_L \arrow[d, "f_I"] \\ I_C \arrow[r, "c_I"'] & I_{NI} \end{tikzcd}$$

initiality of (1):

Given pushout (5) in **PTINets** with $d \in \mathcal{M}$.

$$\begin{tikzcd} B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] \arrow[dr, phantom, "(\mathbf{5})"] & D \arrow[l, "d"'] \arrow[d, "h"] \\ C \arrow[r, "c"'] & NI & E \arrow[l, "e"] \end{tikzcd}$$

The sets T_B and I_B are exactly the boundaries of f_T and f_I, respectively, in **Sets**. So pushouts (3) and (4) are initial and since pushouts in **PTINets** can be constructed componentwise in **Sets** there are pushouts (6) and (7) in **Sets** leading to unique suitable functions $b_T^*, c_T^*, b_I^*, c_I^* \in \mathcal{M}_{\mathbf{Sets}}$ such that (8) and (9) are pushouts in **Sets**.

$$\begin{tikzcd} T_B \arrow[r, "b_T^*"'] \arrow[rr, bend left, "b_T"] \arrow[d, "g_T"'] \arrow[dr, phantom, "(\mathbf{8})"] & T_D \arrow[r, "d_T"'] \arrow[d, "h_T"] \arrow[dr, phantom, "(\mathbf{6})"] & T_L \arrow[d, "f_T"] \\ T_C \arrow[r, "c_T^*"] \arrow[rr, bend right, "c_T"'] & T_E \arrow[r, "e_T"] & NI_T \end{tikzcd} \quad \begin{tikzcd} I_B \arrow[r, "b_I^*"'] \arrow[rr, bend left, "b_I"] \arrow[d, "g_I"'] \arrow[dr, phantom, "(\mathbf{9})"] & I_D \arrow[r, "d_I"'] \arrow[d, "h_I"] \arrow[dr, phantom, "(\mathbf{7})"] & I_L \arrow[d, "f_I"] \\ I_C \arrow[r, "c_I^*"] \arrow[rr, bend right, "c_I"'] & I_E \arrow[r, "e_I"] & NI_I \end{tikzcd}$$

We define a function $c_P^* : P_C \to P_E$ with

$$c_P^*(x) = y \text{ with } e_P(y) = c_P(x)$$

well-definedness of c_P^*:

For the well-definedness of c_P^* we have to show that for every $x \in P_C$ there is a unique $y \in P_E$ with $e_P(y) = c_P(x)$.

Let $x \in P_C$. We use in the following the fact, that from pushout (5) in **PTINets** follows that (10) is a pushout in **Sets**.

$$\begin{tikzcd} P_D \arrow[r, "d_P"] \arrow[d, "h_P"'] \arrow[dr, phantom, "(\mathbf{10})"] & P_L \arrow[d, "f_P"] \\ P_C \arrow[r, "e_P"'] & P_{NI} \end{tikzcd}$$

Case 1: $x \notin f_P(P_L)$
Since (10) is pushout in **Sets** the functions f_P and e_P are jointly surjective. So $x \notin f_P(P_L)$ implies $x \in e_P(P_E)$ and hence there exists $y \in P_E$ with $e_P(y) = x$.

Case 2: $x \in f_P(b_P(P_B))$
Then there exists $z \in P_B$ with $f_P(b_P(z)) = x$.

Case 2.1: $z \in IP_P$
Then from the fact that (10) is pushout in **Sets** together with Fact A.7 follows that d_P and f_P satisfy the gluing condition which means that $z \in d_P(P_D)$, i.e. there exists $z' \in P_D$ with $d_P(z') = z$. Let $y = h_P(z')$. Then we have

$$e_P(y) = e_P(h_P(z')) = f_P(d_P(z')) = f_P(z) = x$$

which means that y is a suitable element.

Case 2.2: $z \in DP_T$
This means that there is $t \in T_G \backslash f_T(T_L)$ with $f_P(z) \in ENV_P(t)$. Since (6) is pushout in **Sets** the functions f_T and e_T are jointly surjective which by the fact that $t \notin f_T(T_L)$ implies that $t \in e_T(T_E)$, i.e. there exists $t' \in T_E$ with $e_T(t') = t$.
Then there is $y \in P_E$ with $y \in ENV_P(t')$ and $e_P(y) = f_P(z) = x$ because P/T morphisms preserve pre and post conditions.

Case 2.3: $z \in DP_I$
Then there exists $i \in I_{NI} \setminus f_I(I_L)$ with $f_P(z) = m_{NI}(i)$. Since (7) is pushout in **Sets** the functions f_I and e_I are jointly surjective which due to the fact that $i \notin f_I(I_L)$ implies that there is $i' \in I_E$ with $e_I(i') = i$. Hence we have

$$x = f_P(z) = m_{NI}(i) = m_{NI}(e_I(i')) = e_P(m_E(i'))$$

which means that $y = m_E(i')$ is a suitable place.

Case 2.4: $z \in P_{IP_T}$
This means that there is $t \in IP_T$ with $z \in ENV_P(t)$. By Fact A.7 there is $t \in d_T(T_D)$ because T_E is a pushout complement of d_T and f_T. So there is $t' \in T_D$ with $t = d_T(t')$. Then we have

$$\begin{aligned} e_P^{\oplus}(pre_E(h_T(t'))) &= pre_G(e_T(h_T(t'))) \\ &= pre_G(f_T(d_T(t'))) \\ &= pre_G(f_T(t)) \\ &= f_P^{\oplus}(pre_L(t)) \end{aligned}$$

and analogously

$$e_P^{\oplus}(post_E(h_T(t'))) = f_P^{\oplus}(post_L(t))$$

which means that there is $y \in ENV_P(h_T(t')) \subseteq P_E$ with

$$e_P(y) = f_P(z) = x$$

Case 2.5: $z \in P_{IP_I}$
Then there exists $i \in IP_I$ with $z = m_L(i)$ which by Fact A.7 implies that there is $i' \in I_D$ with $d_I(i') = i$ because I_E is a pushout complement of d_I and f_I.
Then we have

$$\begin{aligned} & e_P(m_E(h_I(i'))) = e_P(h_P(m_D(i'))) = f_P(d_P(m_D(i'))) \\ = \; & f_P(m_L(d_I(i'))) = f_P(m_L(i)) = f_P(z) = x \end{aligned}$$

and hence $y = m_E(h_I(i'))$ is a suitable place.

So for every $x \in P_C$ there is a suitable $y \in P_E$ with $e_P(y) = c_P(x)$. Let us assume that y is not unique, i.e. there is $y' \in P_E$ with $e_P(y') = c_P(x)$. Since $d \in \mathcal{M}$ and pushouts preserve $\mathcal{M}$-morphisms we also have that $e \in \mathcal{M}$ which means that e_P is injective implying that $y = y'$. Hence c_P is well-defined.

morphism c^*:
We define a PTI morphism $c^* = (c^*_P, c^*_T, c^*_I) : C \to E$. In order to show that c^* is a well-defined PTI morphism we have to show that it preserves pre and post domains and markings.
Let $t \in T_C$ and let $t' \in T_E$ with $e_T(t') = c_T(t)$, i.e. $t' = c^*_T(t)$.
Then we have

$$c_P^{\oplus}(pre_C(t)) = pre_{NI}(c_T(t)) = pre_{NI}(e_T(t')) = e_P^{\oplus}(pre_E(t'))$$

which means that for

$$pre_C(t) = \sum_{i=1}^{n} p_i \quad \text{and} \quad pre_E(t') = \sum_{i=1}^{m} p'_i$$

there is $n = m$ because c_P and e_P are injective and therefore also $c_P^{\oplus}$ and $e_P^{\oplus}$ are injective.
So we have

$$\sum_{i=1}^{n} c_P(p_i) = c_P^{\oplus}(pre_C(t)) = e_P^{\oplus}(pre_E(t')) = \sum_{i=1}^{n} e_P(p'_i)$$

which by the definition of c^*_P means that

$$\begin{aligned} & c_P^{*\oplus}(pre_C(t)) = c_P^{*\oplus}(\sum_{i=1}^{n} p_i) = \sum_{i=1}^{n} c^*_P(p_i) \\ = \; & \sum_{i=1}^{n} p'_i = pre_E(t') = pre_E(c^*_T(t)) \end{aligned}$$

The proof that c^* preserves post domains works analogously.
Let $i \in I_C$ and let $i' \in I_E$ with $e_I(i') = c_I(i)$, i.e. $i' = c_I^*(i)$.
Then we have

$$c_P(m_C(i)) = m_{NI}(c_I(i)) = m_{NI}(e_I(i')) = e_P(m_E(i'))$$

which by definition of c_P^* means that

$$c_P^*(m_C(i)) = m_E(i') = m_E(c_I^*(i))$$

Hence c^* is a well-defined PTI morphism.
The fact that $e \circ c^* = c$ follows directly from the definition of c_P^* and the initiality of (3) and (4).

uniqueness of c^*, existence of b^* and pushout:
By Lemma A.1 the morphism c^* is the unique morphism with $c = e \circ c^*$ and there is a unique morphism $b^* : B \to D$ with $b = d \circ b^*$ such that (11) is a pushout in **PTINets**.

$$\begin{tikzcd}
B \arrow[r, "b^*"'] \arrow[d, "g"'] \arrow[rr, bend left, "b"] \arrow[dr, phantom, "\mathbf{(11)}"] & D \arrow[r, "d"'] \arrow[d, "h"] \arrow[dr, phantom, "\mathbf{(5)}"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "e"] & NI
\end{tikzcd}$$

□

B.2. Proof of 2.11

In this section we prove the equivalence of the fact that PTI morphisms l and f satisfy the gluing condition, and the fact that l and f satisfy the categorical gluing condition.

Proof.

If. Let l and f satisfy the categorical gluing condition, i.e. for initial pushout (1) there is a morphism $b^* : B \to K$ with $l \circ b^* = b$.

$$\begin{tikzcd}
B \arrow[r, "b"'] \arrow[d, "g"'] \arrow[rr, bend left, "b^*"] \arrow[dr, phantom, "\mathbf{(1)}"] & L \arrow[d, "f"] & K \arrow[l, "l"] \\
C \arrow[r, "c"'] & NI &
\end{tikzcd}$$

Let $x \in IP \cup DP$. We have to show that $x \in GP = l_P(P_K) \cup l_T(T_K) \cup l_I(I_K)$.

Case 1: $x \in IP_P \cup IP_I \cup DP$
Then there is $x \in P_B$ and $y \in P_K$ with $b^*(x) = y$ and

$$l(y) = l(b^*(x)) = b(x) = x$$

which means that $x \in l_P(P_K) \subseteq GP$.

Case 2: $x \in IP_T$

Then there is $x \in T_B$ and $y \in T_K$ with $b^*(x) = y$ and

$$l(y) = l(b^*(x)) = b(x) = x$$

which means that $x \in l_T(T_K) \subseteq GP$.

Case 3: $x \in IP_I$

Analogously to Case 2.

Hence l and f satisfy the gluing condition in **PTINets**.

Only If. Let l and f satisfy the gluing condition in **PTINets**, i.e. there is

$$IP \cup DP \subseteq GP.$$

This means that there is $IP_P \subseteq l_P(P_K)$, $IP_T \subseteq l_T(T_K)$ and $IP_I \subseteq l_I(I_K)$ implying that l_P, f_P and l_T, f_T and l_I, f_I satisfy the gluing condition in **Sets**.
From Fact A.7 follows that there are functions $b_P^* : P_B \to P_K$ with $l_P \circ b_P^* = b_P$, $b_T^* : T_B \to T_K$ with $l_T \circ b_T^* = b_T$, and $l_I : I_B \to I_K$ with $l_I \circ b_I^* = b_I$.
We define a PTI morphism $b^* = (b_P^*, b_T^*, b_I^*)$. For the well-definedness we have to show that b^* preserves pre and post domains as well as markings.
Let $t \in T_B$. Then there is

$$\begin{aligned} l_P^{\oplus}(b_P^{*\oplus}(pre_B(t))) &= (l_P \circ b_P^*)^{\oplus}(pre_B(t)) \\ &= b_P^{\oplus}(pre_B(t)) \\ &= pre_L(b_T(t)) \\ &= pre_L(l_T(b_T^*(t))) \\ &= l_P^{\oplus}(pre_K(b_T^*(t))) \end{aligned}$$

and since $\_^{\oplus}$ preserves monomorphisms in **Sets** because monomorphisms in **Sets** are exactly the coequalizers in **Sets** and $\_^{\oplus}$ is a free functor, the morphism $l_P^{\oplus}$ is a monomorphism implying that

$$b_P^{*\oplus}(pre_B(t)) = pre_K(b_T^*(t))$$

The proof that b^* preserves post domains works analogously.
Let $i \in I_B$. Then there is

$$\begin{aligned} l_P(b_P^*(m_B(i))) &= b_P(m_B(i)) \\ &= m_L(b_I(i)) \\ &= m_L(l_I(b_I^*(i))) \\ &= l_P(m_K(b_I^*(i))) \end{aligned}$$

which by the fact that l_P is a monomorphism implies that

$$b_P^*(m_B(i)) = m_K(b_I^*(i))$$

Hence b^* is a well-defined PTI morphism which means that l and f satisfy the categorical gluing condition.

□

B.3. Proof of Fact 3.10

In this section we prove the well-definedness of the AHLI net C defined in Fact 3.10 and that the construction leads to an initial pushout in the category **AHLINets**.

Proof.

well-definedness of C:

$pre_C, post_C : T_C \to (T_{OP_C}(X_C) \otimes P_C)^{\oplus}$:

Let $t \in T_C$ and let $(term, p) \leq pre_C(t)$. Then there is

$$term \in T_{OP_{ANI}}(X_{ANI})_{type_{ANI}(p)} = T_{OP_C}(X_C)_{type_C(p)}.$$

It remains to show that $p \in P_C$.
Due to the fact that $t \in (T_{ANI} \setminus f_T(T_L)) \cup f_T(b_T(T_B))$ we can distinguish the following cases:

Case 1: $t \notin f_T(T_L)$

Case 1.1: $\exists p' \in P_L : f_P(p') = p$
Then due to the fact that there is $(term, p) \leq pre_{ANI}(t)$ there is $p' \in DP_T \subseteq P_B$ which means that $p = f_P(b_P(p')) \in P_C$.

Case 1.2: $\nexists p' \in P_L : f_P(p') = p$
This means that $p \in P_{ANI} \setminus f_P(P_L) \subseteq P_C$.

Case 2: $t \in f_T(b_T(T_B))$
Then there exists $t' \in T_B$ with $f_T(b_T(t')) = t$ and since AHL morphisms preserve pre conditions there is

$$\begin{aligned} pre_{ANI}(t) &= pre_{ANI}(f_T(b_T(t'))) \\ &= (f_\Sigma^\sharp \otimes f_P)^{\oplus}(pre_L(b_T(t'))) \\ &= (f_\Sigma^\sharp \otimes f_P)^{\oplus}((b_\Sigma^\sharp \otimes b_P)^{\oplus}(pre_B(t'))) \end{aligned}$$

By Lemma A.2 there is

$$pre_{ANI}(t) = ((f_\Sigma \circ b_\Sigma)^\sharp \otimes (f_P \circ b_P))^{\oplus}(pre_B(t'))$$

which for $(term, p) \leq pre_{ANI}(t)$ means that

$$(term, p) \leq ((f_\Sigma \circ b_\Sigma)^\sharp \otimes (f_P \circ b_P))^{\oplus}(pre_B(t'))$$

and hence $p \in f_P(b_P(P_B)) \subseteq P_C$.

The proof for $post_C$ works analogously.

$m_C : I_C \to A_C \otimes P_C$:

Let $i \in I_C$ and let $(a, p) = m_C(i)$. Then there is

$$a \in A_{ANI, type_{ANI}(p)} = A_{C, type_C(p)}.$$

It remains to show that $p \in P_C$. Due to the fact that

$$i \in (I_{ANI} \setminus f_I(I_L)) \cup f_I(b_I(I_B))$$

we can distinguish the following cases:

Case 1: $i \notin f_I(I_L)$

Case 1.1: $\exists p' \in P_L : f_P(p') = p$
This means that $p' \in DP$ and thus $p' \in P_B$ and $p = f_P(b_P(p')) \in P_C$.

Case 1.2: $\nexists p' \in P_L : f_P(p) = p$
This means that $p \notin f_P(P_L)$, i.e. $p \in ANI \setminus f_P(P_L)$ and hence $p \in P_C$.

Case 2: $i \in f_I(b_I(I_B))$
Then there exists $j \in I_B$ with $f_I(b_I(j)) = i$ and we have

$$\begin{aligned} m_C(i) &= m_{ANI}(i) \\ &= m_{ANI}(f_I(b_I(j))) \\ &= (f_A \otimes f_P)(m_L(b_I(j))) \\ &= (f_A \otimes f_P)((b_A \otimes b_P)(m_B(j))) \end{aligned}$$

which means that $p \in f_P(b_P(P_B))$ and hence $p \in P_C$.

well-definedness of c:
We obtain an inclusion morphism $c : C \to ANI$ from the fact that pre_C, $post_C$, $cond_C$, $type_C$ and m_C are restrictions of the respective functions in ANI. Furthermore there is $c_\Sigma = id_{\Sigma_C}$ and $c_A = id_{A_C}$ which are well-defined signature and algebra morphisms, respectively.

well-definedness of g:
For $J = \{P, T, I\}$ we obtain well-defined functions $g_J : J_B \to J_C$ because for $j \in J_B$ there is

$$g_J(j) = f_J(j) = f_J(b_J(j)) \in J_C$$

The morphism g preserves pre and post domains, conditions, types and markings because it is a restriction of f which is a well-defined AHLI morphism.
Furthermore there is $g_\Sigma = f_\Sigma$ and $g_A = f_A$.

(1) is pushout:
Due to Lemma A.6 the diagrams (2)-(4) are pushouts in **Sets**.

$$\begin{tikzcd} P_B \arrow[r, "b_P"] \arrow[d, "g_P"'] \arrow[dr, phantom, "(\mathbf{2})"] & P_L \arrow[d, "f_P"] \\ P_C \arrow[r, "c_P"'] & P_{ANI} \end{tikzcd} \quad \begin{tikzcd} T_B \arrow[r, "b_T"] \arrow[d, "g_T"'] \arrow[dr, phantom, "(\mathbf{3})"] & T_L \arrow[d, "f_T"] \\ T_C \arrow[r, "c_T"'] & T_{ANI} \end{tikzcd} \quad \begin{tikzcd} I_B \arrow[r, "b_I"] \arrow[d, "g_I"'] \arrow[dr, phantom, "(\mathbf{4})"] & I_L \arrow[d, "f_I"] \\ I_C \arrow[r, "c_I"'] & I_{ANI} \end{tikzcd}$$

Moreover diagram (5) is pushout in **Sig** and (6) is pushout in **Algs**.

$$\begin{tikzcd}
\Sigma_L \arrow[r, "id_{\Sigma_L}"] \arrow[d, "f_\Sigma"'] \arrow[dr, phantom, "(\mathbf{5})"] & \Sigma_L \arrow[d, "f_\Sigma"] \\
\Sigma_{ANI} \arrow[r, "id_{\Sigma_{ANI}}"'] & \Sigma_{ANI}
\end{tikzcd}
\qquad
\begin{tikzcd}
(\Sigma_L, A_L) \arrow[r, "id_{(\Sigma_L, A_L)}"] \arrow[d, "(f_\Sigma, f_A)"'] \arrow[dr, phantom, "(\mathbf{6})"] & (\Sigma_L, A_L) \arrow[d, "(f_\Sigma, f_A)"] \\
(\Sigma_{ANI}, A_{ANI}) \arrow[r, "id_{(\Sigma_{ANI}, A_{ANI})}"'] & (\Sigma_{ANI}, A_{ANI})
\end{tikzcd}$$

The pushouts (2)-(6) imply that (1) is pushout in **AHLINets** because the pushout in **AHLINets** can be constructed componentwise.

initiality of (1):
Given pushout (7) in **AHLINets** with $d \in \mathcal{M}$.

$$\begin{tikzcd}
B \arrow[r, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] \arrow[dr, phantom, "(\mathbf{7})"] & D \arrow[l, "d"'] \arrow[d, "h"] \\
C \arrow[r, "c"'] & ANI & E \arrow[l, "e"]
\end{tikzcd}$$

The sets T_B and I_B are exactly the boundaries of f_T and f_I, respectively, in **Sets**. So pushouts (3) and (4) are initial and since pushouts in **AHLINets** can be constructed componentwise in **Sets** there are pushouts (8) and (9) in **Sets** leading to unique suitable functions $b_T^*, c_T^*, b_I^*, c_I^* \in \mathcal{M}_{\mathbf{Sets}}$ such that (10) and (11) are pushouts in **Sets**.

$$\begin{tikzcd}
T_B \arrow[r, "b_T^*"'] \arrow[rr, bend left, "b_T"] \arrow[d, "g_T"'] \arrow[dr, phantom, "(\mathbf{10})"] & T_D \arrow[r, "d_T"'] \arrow[d, "h_T"] \arrow[dr, phantom, "(\mathbf{8})"] & T_L \arrow[d, "f_T"] \\
T_C \arrow[r, "c_T^*"] \arrow[rr, bend right, "c_T"'] & T_E \arrow[r, "e_T"] & ANI_T
\end{tikzcd}
\qquad
\begin{tikzcd}
I_B \arrow[r, "b_I^*"'] \arrow[rr, bend left, "b_I"] \arrow[d, "g_I"'] \arrow[dr, phantom, "(\mathbf{11})"] & I_D \arrow[r, "d_I"'] \arrow[d, "h_I"] \arrow[dr, phantom, "(\mathbf{9})"] & I_L \arrow[d, "f_I"] \\
I_C \arrow[r, "c_I^*"] \arrow[rr, bend right, "c_I"'] & I_E \arrow[r, "e_I"] & ANI_I
\end{tikzcd}$$

function c_P^*:
We define a function $c_P^* : P_C \to P_E$ with

$$c_P^*(x) = y \text{ with } e_P(y) = c_P(x)$$

For the well-definedness of c_P^* we have to show that for every $x \in P_C$ there is a unique $y \in P_E$ with $e_P(y) = c_P(x)$.
Let $x \in P_C$. We need in the following that pushout (7) in **AHLINets** implies pushout (12) in **Sets**.

$$\begin{tikzcd}
P_D \arrow[r, "d_P"] \arrow[d, "h_P"'] \arrow[dr, phantom, "(\mathbf{12})"] & P_L \arrow[d, "f_P"] \\
P_E \arrow[r, "e_P"'] & P_{ANI}
\end{tikzcd}$$

Case 1: $x \notin f_P(P_L)$
Since (12) is pushout in **Sets** the functions f_P and e_P are jointly surjective. So $x \notin f_P(P_L)$ implies $x \in e_P(P_E)$ and hence there exists $y \in P_E$ with $e_P(y) = x$.

Case 2: $x \in f_P(b_P(P_B))$
Then there exists $z \in P_B$ with $f_P(b_P(z)) = x$.

Case 2.1: $z \in IP_P$
Then from the fact that (12) is pushout in **Sets** together with Fact A.7 follows that d and f satisfy the gluing condition which means that $z \in d_P(P_D)$, i.e. there exists $z' \in P_D$ with $d_P(z') = z$. Let $y = h_P(z')$. Then we have

$$e_P(y) = e_P(h_P(z')) = f_P(d_P(z')) = f_P(z) = x$$

which means that y is a suitable element.

Case 2.2: $z \in DP_T$
This means that there is $t \in T_{ANI} \setminus f_T(T_L)$ with $f_P(z) \in ENV_P(t)$. Since (8) is pushout in **Sets** the functions f_T and e_T are jointly surjective which by the fact that $t \notin f_T(T_L)$ implies that $t \in e_T(T_E)$, i.e. there exists $t' \in T_E$ with $e_T(t') = t$.
Then there is $y \in P_E$ with $y \in ENV_P(t')$ and $e_P(y) = f_P(z) = x$ because AHLI morphisms preserve pre and post conditions.

Case 2.3: $z \in DP_I$
Then there exists $i \in I_{ANI} \setminus f_I(I_L)$ with $f_P(z) = \pi_P(m_{ANI}(i))$. Since (9) is pushout in **Sets** the functions f_I and e_I are jointly surjective which due to the fact that $i \notin f_I(I_L)$ implies that there is $i' \in I_E$ with $e_I(i') = i$. Hence we have

$$\begin{aligned} x = f_P(z) &= \pi_P(m_{ANI}(i)) \\ &= \pi_P(m_{ANI}(e_I(i'))) \\ &= \pi_P((e_A \otimes e_P)(m_E(i'))) \\ &= e_P(m_E(i')) \end{aligned}$$

which means that $y = m_E(i')$ is a suitable place.

Case 2.4: $z \in P_{IP_T}$
This means that there is $t \in IP_T$ with $z \in ENV_P(t)$. By Fact A.7 there is $t \in d_T(T_D)$ because T_E is a pushout complement of d_T and f_T. So there is $t' \in T_D$ with $t = d_T(t')$. Then we have

$$\begin{aligned} (e_\Sigma^\sharp \otimes e_P)^\oplus(pre_E(h_T(t'))) &= pre_{ANI}(e_T(h_T(t'))) \\ &= pre_{ANI}(f_T(d_T(t'))) \\ &= pre_{ANI}(f_T(t)) \\ &= (f_\Sigma^\sharp \otimes f_P)^\oplus(pre_L(t)) \end{aligned}$$

and analogously

$$(e_\Sigma^\sharp \otimes e_P)^\oplus(post_E(h_T(t'))) = (f_\Sigma^\sharp \otimes f_P)^\oplus(post_L(t))$$

which means that there is $y \in ENV_P(h_T(t')) \subseteq P_E$ with

$$e_P(y) = f_P(z) = x$$

Case 2.5: $z \in P_{IP_I}$
Then there exists $i \in IP_I$ with $z = \pi_P(m_L(i))$ which by Fact A.7 implies that there is $i' \in I_D$ with $d_I(i') = i$ because I_E is a pushout complement of d_I and f_I.
Then we have

$$\begin{aligned}(e_A \otimes e_P)(m_E(h_I(i'))) &= m_{ANI}(e_I(h_I(i'))) \\ &= m_{ANI}(f_I(d_I(i'))) \\ &= (f_A \otimes f_P)(m_L(i))\end{aligned}$$

which means that there is $y = \pi_P(m_E(h_I(i'))) \subseteq P_E$ with

$$e_P(y) = f_P(z) = x$$

So for every $x \in P_C$ there is a suitable $y \in P_E$ with $e_P(y) = c_P(x)$. Let us assume that y is not unique, i.e. there is $y' \in P_E$ with $e_P(y') = c_P(x)$. Since $d \in \mathcal{M}$ and pushouts preserve $\mathcal{M}$-morphisms there is also $e \in \mathcal{M}$ which means that e_P is injective implying that $y = y'$. Hence c_P is well-defined.

signature morphism c_Σ^*:
Due to the fact that $e \in \mathcal{M}$ the signature morphism e_Σ is an isomorphism. We define

$$c_\Sigma^* = e_\Sigma^{-1}$$

Since $\Sigma_C = \Sigma_{ANI}$ the morphism c_Σ^* is well-defined.

algebra morphism c_A^*:
Also the algebra morphism e_A is an isomorphism because $e \in \mathcal{M}$. So we define

$$c_A^* = e_A^{-1}$$

leading to a well-defined algebra morphism because $A_C = A_{ANI}$.

morphism c^*:
We define an AHLI morphism $c^* = (c_\Sigma^*, c_P^*, c_T^*, c_A^*, c_I^*) : C \to E$. In order to show that c^* is a well-defined AHLI morphism we have to show that it preserves pre and post conditions, conditions, types and markings.

types:

Let $p \in P_C$ and let $p' \in P_E$ with $c_P(p) = e_P(p')$, i.e. $c^*_P(p) = p'$.
Furthermore let $c^*_\Sigma = (c^*_S, c^*_{OP})$ and $e_\Sigma = (e_S, e_{OP})$. Then we have

$$\begin{aligned} c^*_S(type_C(p)) &= e_S^{-1}(type_C(p)) \\ &= e_S^{-1}(type_{ANI}(p)) \\ &= e_S^{-1}(type_{ANI}(c_P(p))) \\ &= e_S^{-1}(type_{ANI}(e_P(p'))) \\ &= e_S^{-1}(e_S(type_E(p'))) \\ &= type_E(p') \\ &= type_E(c^*_P(p)) \end{aligned}$$

pre and post conditions:

Let $t \in T_C$.
Due to the definition of c^*_P there is $c_P = e_P \circ c^*_P$. So we have

$$\begin{aligned} & (e^\sharp_\Sigma \otimes e_P)^\oplus((c^{*\sharp}_\Sigma \otimes c^*_P)^\oplus(pre_C(t))) \\ = \;& (e^\sharp_\Sigma \otimes e_P)^\oplus((c^{*\sharp}_\Sigma \otimes c^*_P)^\oplus(pre_C(t))) \\ \overset{\text{Lemma } A.2}{=} \;& ((e_\Sigma \circ c^*_\Sigma)^\sharp \otimes (e_P \circ c^*_P))^\oplus(pre_C(t)) \\ = \;& ((e_\Sigma \circ e_\Sigma^{-1})^\sharp \otimes c_P)^\oplus(pre_C(t)) \\ = \;& ((id_{\Sigma_{ANI}})^\sharp \otimes c_P)^\oplus(pre_C(t)) \\ = \;& (c^\sharp_\Sigma \otimes c_P)^\oplus(pre_C(t)) \\ = \;& pre_{ANI}(c_T(t)) \\ = \;& pre_{ANI}(e_T \circ c^*_T(t)) \\ = \;& (e^\sharp_\Sigma \otimes e_P)^\oplus(pre_E(c^*_T(t))) \end{aligned}$$

Since $e \in \mathcal{M}$ is a monomorphism and $\_ \otimes \_$, $\_^\sharp$ and $\_^\oplus$ preserve monomorphisms, there is also $(e^\sharp_\Sigma \otimes e_P)^\oplus$ a monomorphism. So the above equation implies

$$(c^{*\sharp}_\Sigma \otimes c^*_P)^\oplus(pre_C(t)) = pre_E(c^*_T(t))$$

The proof that c^* preserves post conditions works analogously.

conditions:

Let $t \in T_C$ and let $t' \in T_E$ with $e_T(t') = c_T(t)$, i.e. $t' = c^*_T(t)$.
Due to the fact that $c_\Sigma = id_{\Sigma_C}$ there is

$$\begin{aligned} cond_C(t) &= \mathcal{P}_{fin}(c^\sharp_\Sigma)(cond_C(t)) \\ &= cond_{ANI}(c_T(t)) \\ &= cond_{ANI}(e_T(t')) \\ &= \mathcal{P}_{fin}(e^\sharp_\Sigma)(cond_E(t')) \end{aligned}$$

which by the definition of c^*_Σ implies that

$$\begin{array}{rcl}
\mathcal{P}_{fin}(c^{*\sharp}_\Sigma)(cond_C(t)) & = & \mathcal{P}_{fin}(c^{*\sharp}_\Sigma)(\mathcal{P}_{fin}(e^\sharp_\Sigma)(cond_E(t'))) \\
& \overset{\text{Lemma } A.3}{=} & \mathcal{P}_{fin}((c^*_\Sigma \circ e_\Sigma)^\sharp)(cond_E(t')) \\
& = & \mathcal{P}_{fin}((e^{-1}_\Sigma \circ e_\Sigma)^\sharp)(cond_E(t')) \\
& = & \mathcal{P}_{fin}(id^\sharp_{\Sigma_E})(cond_E(t')) \\
& = & \mathcal{P}_{fin}(id_{T_{OP_E}(X_E)})(cond_E(t')) \\
& = & cond_E(t') \\
& = & cond_E(c^*_T(t))
\end{array}$$

markings:

Let $i \in I_C$ and $m_C(i) = (a, p)$.
Then we have

$$\begin{array}{rcl}
(e_A \otimes e_P)((c^*_A \otimes c^*_P)(m_C(i))) & = & (e_A \otimes e_P)((c^*_A \otimes c^*_P)(a, p)) \\
& = & (e_A \otimes e_P)(c^*_{A,type_C(p)}(a), c^*_P(p)) \\
& = & (e_{A,type_E(c^*_P(p))}(c^*_{A,type_C(p)}(a)), e_P(c^*_P(p))) \\
& = & (e_{A,c^*_S(type_C(p))}(c^*_{A,type_C(p)}(a)), e_P(c^*_P(p))) \\
& = & (V_{c^*_\Sigma}(e_A)_{type_C(p)}(c^*_{A,type_C(p)}(a)), e_P(c^*_P(p))) \\
& = & (V_{c^*_\Sigma}(e_A)_{type_C(p)}(e^{-1}_{A,type_C(p)}(a), c_P(p))) \\
& = & ((V_{c^*_\Sigma}(e_A) \circ e^{-1}_A)_{type_C(p)}(a), c_P(p)) \\
& = & (a, c_P(p)) \\
& = & (c_A(a), c_P(p)) \\
& = & (c_A \otimes c_P)(a, p) \\
& = & (c_A \otimes c_P)(m_C(i)) \\
& = & m_{ANI}(c_I(i)) \\
& = & m_{ANI}(e_I \circ c^*_I(i)) \\
& = & (e_A \otimes e_P)(m_E(c^*_I(i)))
\end{array}$$

and since $e \in \mathcal{M}$ is a monomorphism and $\_ \otimes \_$ preserves monomorphisms, there is also $(e_A \otimes e_P)$ a monomorphism. So the equation above implies

$$(c^*_A \otimes c^*_P)(m_C(i)) = m_E(c^*_I(i))$$

Hence c^* is a well-defined AHLI morphism.
The fact that $e \circ c^* = c$ follows directly from the definitions of c^*_P, c^*_Σ and c^*_A and the initiality of (3) and (4).

uniqueness of c^*, existence of b^* and pushout:

By Lemma A.1 the morphism c^* is the unique morphism with $c = e \circ c^*$ and there is a unique morphism $b^* : B \to D$ with $b = d \circ b^*$ such that (13) is a pushout in **AHLINets**.

$$\begin{tikzcd}
B \arrow[r, "b^*"'] \arrow[rr, bend left, "b"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{13})"] & D \arrow[r, "d"'] \arrow[d, "h"] \arrow[dr, phantom, "(\mathbf{7})"] & L \arrow[d, "f"] \\
C \arrow[r, "c^*"] \arrow[rr, bend right, "c"'] & E \arrow[r, "e"] & ANI
\end{tikzcd}$$

□

B.4. Proof of Fact 3.11

In this section we prove that the fact that AHLI morphisms f and l satisfy the gluing condition in **AHLINets** is equivalent to the fact that f and l satisfy the categorical gluing condition.

Proof.

If. Let l and f satisfy the categorical gluing condition, i.e. for initial pushout (1) there is a morphism $b^* : B \to K$ with $l \circ b^* = b$.

$$\begin{tikzcd}
B \arrow[r, "b"'] \arrow[rr, bend left, "b^*"] \arrow[d, "g"'] \arrow[dr, phantom, "(\mathbf{1})"] & L \arrow[d, "f"] & K \arrow[l, "l"] \\
C \arrow[r, "c"'] & ANI &
\end{tikzcd}$$

Let $x \in IP \cup DP$. We have to show that $x \in GP = l_P(P_K) \cup l_T(T_K) \cup l_I(I_K)$. The proof works completely analogously to the proof in Fact 2.11. Hence l and f satisfy the gluing condition in **AHLINets**.

Only If. Let l and f satisfy the gluing condition in **AHLINets**, i.e. there is

$$IP \cup DP \subseteq GP.$$

This means that there is $IP_P \subseteq l_P(P_K)$, $IP_T \subseteq l_T(T_K)$ and $IP_I \subseteq l_I(I_K)$ implying that l_P, f_P and l_T, f_T and l_I, f_I satisfy the gluing condition in **Sets**.
From Fact A.7 follows that there are functions $b_P^* : P_B \to P_K$ with $l_P \circ b_P^* = b_P$, $b_T^* : T_B \to T_K$ with $l_T \circ b_T^* = b_T$, and $l_I : I_B \to I_K$ with $l_I \circ b_I^* = b_I$.
We define an AHLI morphism $b^* = (b_P^*, b_T^*, b_\Sigma^*, b_A^*, b_I^*)$ with $b_\Sigma^* = l_\Sigma^{-1}$ and $b_A^* = l_A^{-1}$. The signature morphism l_Σ^{-1} and algebra morphism l_A^{-1} exist because $l \in \mathcal{M}$ and hence l_Σ and l_A are isomorphisms.
For the well-definedness of b^* it remains to show that b^* preserves pre and post conditions, conditions, types, and markings.

types:
Let $p \in P_B$ and let $p' \in P_K$ with $b_P(p) = l_P(p')$, i.e. $b_P^*(p) = p'$.

Then we have

$$
\begin{aligned}
b_S^*(type_C(p)) &= b_S^*(type_{ANI}(p)) \\
&= b_S^*(type_{ANI}(c_P(p))) \\
&= b_S^*(type_{ANI}(e_P(p'))) \\
&= b_S^*(e_S(type_E(p'))) \\
&= e_S^{-1}(e_S(type_E(p'))) \\
&= type_E(p') \\
&= type_E(c_P^*(p))
\end{aligned}
$$

pre and post conditions:

Let $t \in T_B$. Then there is

$$
\begin{aligned}
& (l_\Sigma^\sharp \otimes l_P)^\oplus((b_\Sigma^{*\sharp} \otimes b_P^*)^\oplus(pre_B(t))) \\
\overset{\text{Lemma } A.2}{=} \quad & ((l_\Sigma \circ b_\Sigma^*)^\sharp \otimes (l_P \circ b_P^*))^\oplus(pre_B(t)) \\
= \quad & ((l_\Sigma \circ b_\Sigma^*)^\sharp \otimes (l_P \circ b_P^*))^\oplus(pre_B(t)) \\
= \quad & ((l_\Sigma \circ l_\Sigma^{-1})^\sharp \otimes (l_P \circ b_P^*))^\oplus(pre_B(t)) \\
= \quad & (id_{\Sigma_L}^\sharp \otimes b_P)^\oplus(pre_B(t)) \\
= \quad & (b_\Sigma^\sharp \otimes b_P)^\oplus(pre_B(t)) \\
= \quad & (pre_{ANI}(b_T(t))) \\
= \quad & (pre_{ANI}(l_T \circ b_T^*(t))) \\
= \quad & (l_\Sigma^\sharp \otimes l_P)^\oplus(pre_K(b_T^*(t)))
\end{aligned}
$$

Since $l \in \mathcal{M}$ is a monomorphism and $\_ \otimes \_$, $\_^\sharp$ and $\_^\oplus$ preserve monomorphisms, there is also $(l_\Sigma^\sharp \otimes l_P)^\oplus$ a monomorphism. So the above equation implies

$$
(b_\Sigma^{*\sharp} \otimes b_P^*)^\oplus(pre_B(t)) = pre_K(b_T^*(t))
$$

The proof for the post conditions works analogously.

conditions:

Let $t \in T_B$ and let $t' \in T_K$ with $b_T(t) = l_T(t')$, i.e. $t' = b_T^*(t)$. Then we have

$$\begin{array}{rcl}
\mathcal{P}_{fin}(b_\Sigma^{*\sharp})(cond_B(t)) & = & \mathcal{P}_{fin}(b_\Sigma^{*\sharp})(cond_L(t)) \\
& = & \mathcal{P}_{fin}(b_\Sigma^{*\sharp})(cond_L(b_T(t))) \\
& = & \mathcal{P}_{fin}(b_\Sigma^{*\sharp})(cond_L(l_T(t'))) \\
& = & \mathcal{P}_{fin}(b_\Sigma^{*\sharp})(\mathcal{P}_{fin}(l_\Sigma^\sharp)(cond_K(t'))) \\
& \overset{\text{Lemma } A.3}{=} & \mathcal{P}_{fin}((b_\Sigma^* \circ l_\Sigma)^\sharp)(cond_K(t')) \\
& = & \mathcal{P}_{fin}((l_\Sigma^{-1} \circ l_\Sigma)^\sharp)(cond_K(t')) \\
& = & \mathcal{P}_{fin}(id_{\Sigma_K})^\sharp)(cond_K(t')) \\
& = & cond_K(t') \\
& = & cond_K(b_T^*(t))
\end{array}$$

markings:

Let $i \in I_B$ and $m_B(i) = (a, p)$.
Then we have

$$\begin{array}{rcl}
(l_A \otimes l_P)((b_A^* \otimes b_P^*)(m_B(i))) & = & (l_A \otimes l_P)((b_A^* \otimes b_P^*)(a, p)) \\
& = & (l_A \otimes l_P)(b_{A,type_B(p)}^*(a), b_P^*(p)) \\
& = & (l_{A,type_K(b_P^*(p))}(b_{A,type_B(p)}^*(a)), l_P(b_P^*(p))) \\
& = & (l_{A,b_S^*(type_B(p))}(b_{A,type_B(p)}^*(a)), l_P(b_P^*(p))) \\
& = & (V_{b_\Sigma^*}(l_A)_{type_B(p)}(b_{A,type_B(p)}^*(a)), l_P(b_P^*(p))) \\
& = & (V_{b_\Sigma^*}(l_A)_{type_B(p)}(l_{A,type_B(p)}^{-1}(a), b_P(p))) \\
& = & ((V_{b_\Sigma^*}(l_A) \circ l_A^{-1})_{type_B(p)}(a), b_P(p)) \\
& = & (a, b_P(p)) \\
& = & (b_A(a), b_P(p)) \\
& = & (b_A \otimes b_P)(a, p) \\
& = & (b_A \otimes b_P)(m_B(i)) \\
& = & m_{ANI}(b_I(i)) \\
& = & m_{ANI}(l_I \circ b_I^*(i)) \\
& = & (l_A \otimes l_P)(m_K(b_I^*(i)))
\end{array}$$

and since $l \in \mathcal{M}$ is a monomorphism and $\_ \otimes \_$ preserves monomorphisms, there is also $(l_A \otimes l_P)$ a monomorphism. So the equation above implies

$$(b_A^* \otimes b_P^*)(m_B(i)) = m_K(b_I^*(i))$$

Hence b^* is a well-defined AHLI morphism. The required commutativity follows from the commutativity of its components. So l and f satisfy the categorical gluing condition.

$\square$